Dense Orientation Recovery via Lightweight Graph Attention for Real-Time Human Avatar Animation from Markerless Tracking

Deutsches Forschungszentrum für Künstliche Intelligenz
German Research Center for Artificial Intelligence

RPTU

iu INTERNATIONALE HOCHSCHULE

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

[1] German Research Center for Artificial Intelligence (DFKI), Kaiserslautern, Germany
[2] RPTU Kaiserslautern-Landau, Kaiserslautern, Germany
[3] International University of Applied Sciences, Erfurt, Germany

ACM SIGGRAPH Eurographics SCA 2026

1. Problem Statement

- Markerless tracking systems provide 3D joint positions
  - avatar animation requires joint orientations for skinning
- This position-to-orientation problem is under-constrained
  - twist around the bone is not determined by positions alone

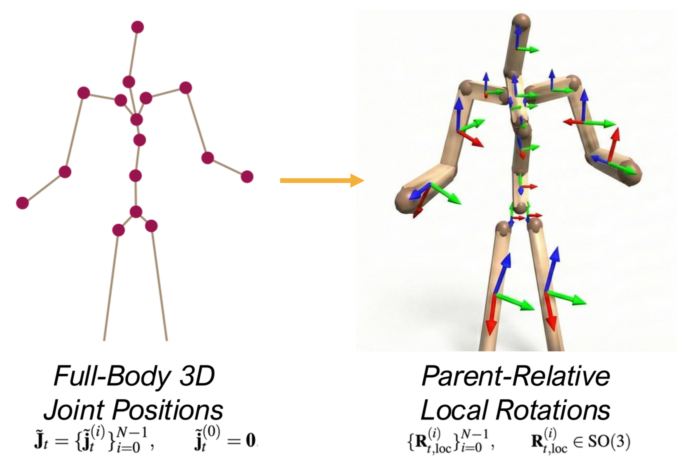


Fig. 1: ***Dense orientation recovery task.*** *Given a set of N 3D joint positions, compute N parent-relative local rotations that transform the 3D rig from rest pose to the articulated pose.*

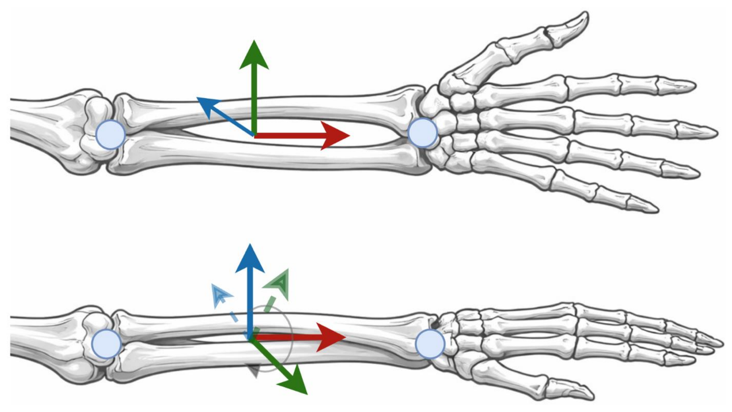


Fig. 2: ***Twist ambiguity problem.*** *Identical joint positions (blue dots) are produced by any rotation around the bone axis (red). This makes the inverse problem ill-posed without strong priors.*

2. Contributions

1. Propose **bone-aligned world rotations** as a canonical, rig-agnostic, stable learning target for neural inverse kinematics
2. Introduce IK-GAT, a **lightweight graph-attention neural network**, to recover joint orientations from positions using a hybrid approach
   - **learned stage** recovers canonical bone-aligned world rotations
   - **analytic stage** converts to rig-specific parent-relative local rotations

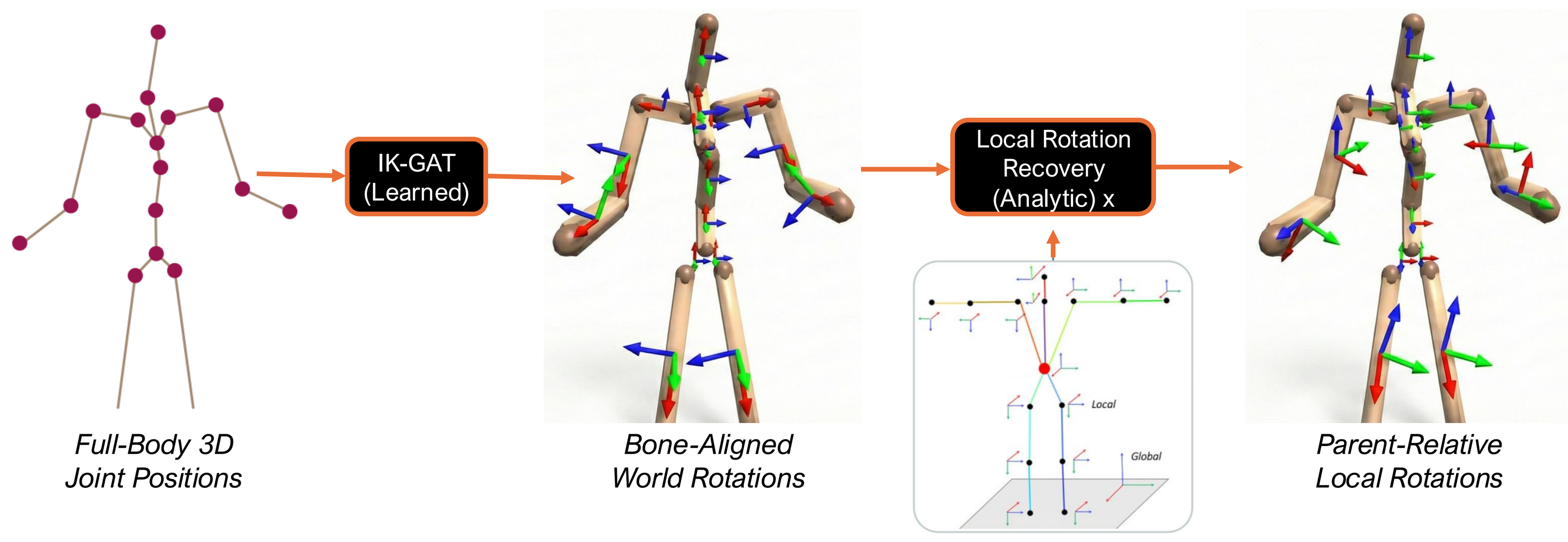


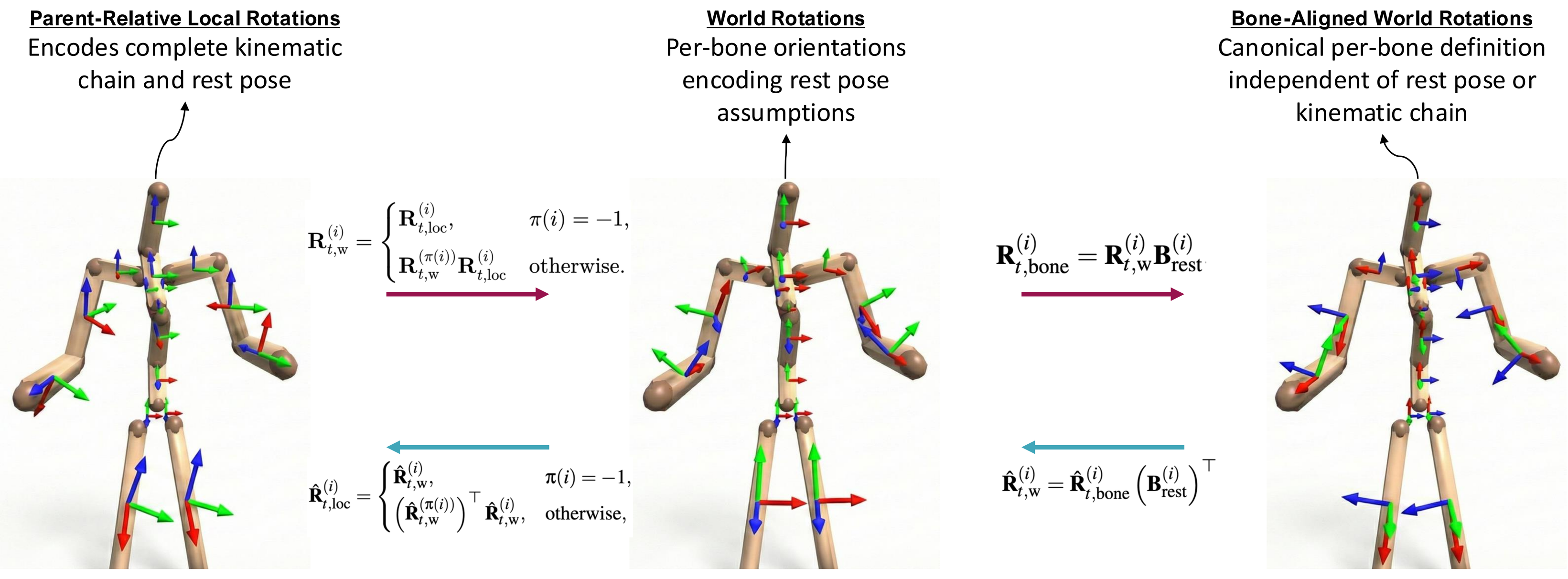
Fig. 3: ***Orientation recovery using IK-GAT.*** *We use a two-stage inference pipeline to first recover canonical world rotations using the learned neural network, followed by an analytical step to convert them to rig-specific local rotations.*

3(a). Method: Bone-Aligned World Rotations

- Derive bone-aligned rest frame from rest pose (once per-rig).

$$\bar{\mathbf{x}}^{(i)} = \frac{\bar{\mathbf{j}}^{(t(i))} - \bar{\mathbf{j}}^{(s(i))}}{\left\|\bar{\mathbf{j}}^{(t(i))} - \bar{\mathbf{j}}^{(s(i))}\right\|_2} \qquad \tilde{\mathbf{y}}^{(i)} = \mathbf{r}^{(i)} - \left\langle \mathbf{r}^{(i)}, \bar{\mathbf{x}}^{(i)} \right\rangle \bar{\mathbf{x}}^{(i)}, \qquad \bar{\mathbf{y}}^{(i)} = \frac{\tilde{\mathbf{y}}^{(i)}}{\|\tilde{\mathbf{y}}^{(i)}\|_2},$$

$$\mathbf{r}^{(i)} = \begin{cases} \mathbf{u}, & i = 0, \\ \bar{\mathbf{y}}^{(\pi(i))}, & \text{otherwise,} \end{cases} \qquad \bar{\mathbf{z}}^{(i)} = \bar{\mathbf{x}}^{(i)} \times \bar{\mathbf{y}}^{(i)}. \qquad \mathbf{B}^{(i)}_{\text{rest}} = \left[\bar{\mathbf{x}}^{(i)} \;\; \bar{\mathbf{y}}^{(i)} \;\; \bar{\mathbf{z}}^{(i)}\right] \in \mathrm{SO}(3),$$



$$\mathbf{R}^{(i)}_{t,\mathrm{w}} = \begin{cases} \mathbf{R}^{(i)}_{t,\mathrm{loc}}, & \pi(i) = -1, \\ \mathbf{R}^{(\pi(i))}_{t,\mathrm{w}} \mathbf{R}^{(i)}_{t,\mathrm{loc}} & \text{otherwise.} \end{cases} \qquad \mathbf{R}^{(i)}_{t,\mathrm{bone}} = \mathbf{R}^{(i)}_{t,\mathrm{w}} \mathbf{B}^{(i)}_{\mathrm{rest}}$$

$$\hat{\mathbf{R}}^{(i)}_{t,\mathrm{loc}} = \begin{cases} \hat{\mathbf{R}}^{(i)}_{t,\mathrm{w}}, & \pi(i) = -1, \\ \left(\hat{\mathbf{R}}^{(\pi(i))}_{t,\mathrm{w}}\right)^{\top} \hat{\mathbf{R}}^{(i)}_{t,\mathrm{w}}, & \text{otherwise,} \end{cases} \qquad \hat{\mathbf{R}}^{(i)}_{t,\mathrm{w}} = \hat{\mathbf{R}}^{(i)}_{t,\mathrm{bone}} \left(\mathbf{B}^{(i)}_{\mathrm{rest}}\right)^{\top}$$

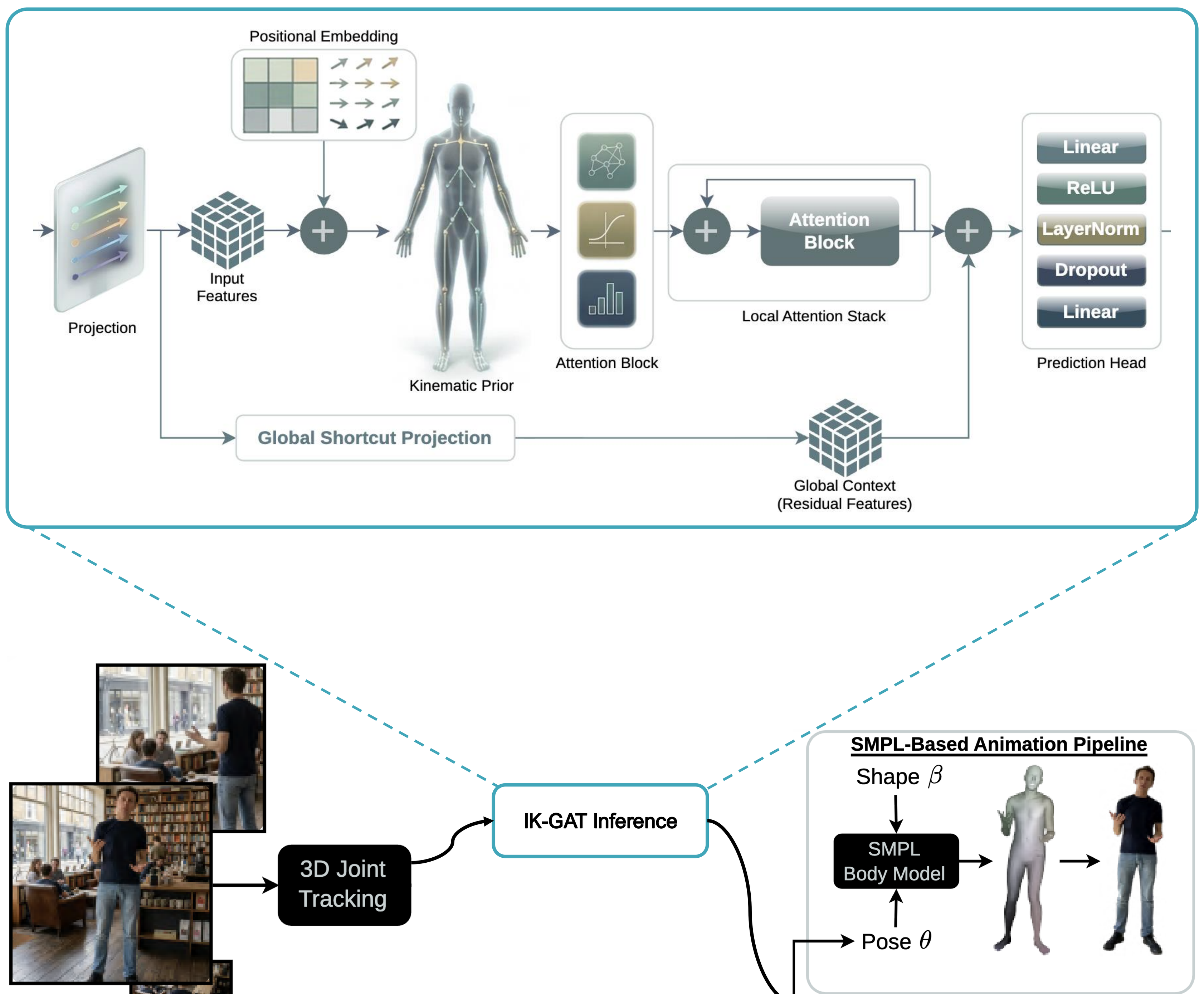
Fig. 4: ***Bone-aligned world rotations.*** *Pre-computed rest frames are used to preprocess articulated local rotations to bone-aligned world representation during training. At inference, the mapping is exactly inverted to rig-specific parent-relative local rotations.*

- Bone-aligned rotation targets during training improve performance:

Rotation Space	MPJAE (deg)	MPJPE (mm)	P-MPJPE (mm)	MPVE (mm)	P-MPVE (mm)
Parent-Relative	7.93	31.68	23.49	47.31	29.29
Bone-World	7.43	18.60	12.80	24.47	18.08
Improvement	0.50 ▼	13.08 ▼	10.69 ▼	22.84 ▼	11.21 ▼

3(b). Method: IK-GAT Architecture

- IK-GAT uses skeletal graph attention with kinematic priors, a continuous 6D rotation representation, and geodesic supervision on SO(3).



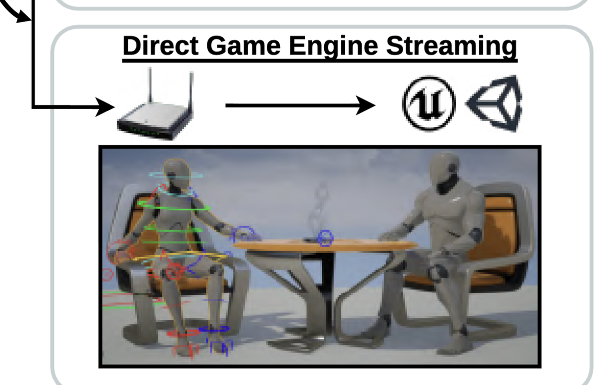
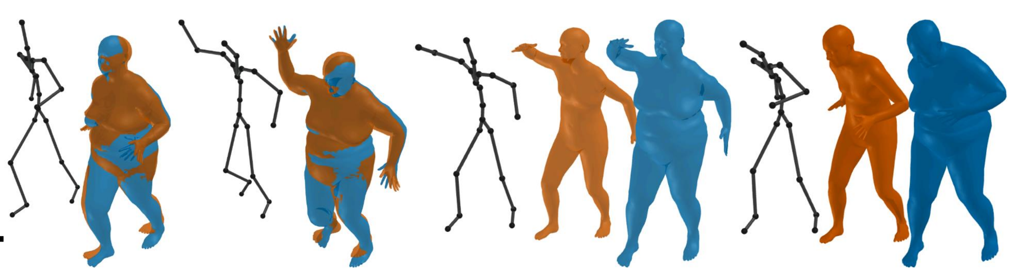


Fig. 6: ***IK-GAT architecture and deployment.*** *In deployment, IK-GAT acts as a connecting module between 3D joint tracking and downstream animation applications. Tracked joints (e.g., from markerless body tracking) can be converted to fully-articulated rotations encoding both swing and twist. These rotations can drive parametric body models like SMPL or rigged avatars in game engines to produce high-quality animations with minimal manual setup.*

4(a). Results: SMPL Benchmark

- IK-GAT reduces test MPJAE from 9.06° for a dense Transformer to 7.43° while using 8.6× fewer parameters.



Method	Representation	MPJAE	MPJPE	P-MPJPE	MPVE	P-MPVE	Params
Multi-Layer Perceptron (MLP)	6D	39.32	211.70	117.86	299.64	130.95	2.11
Graph Convolutional Network (GCN)	Quaternions	14.72	48.18	36.37	57.44	37.76	0.13
Graph Convolutional Network (GCN)	6D	14.31	37.94	31.78	44.19	30.51	0.13
Spatio-Temporal Convolutional Network	6D	12.57	43.05	34.51	44.93	31.90	1.32
Transformer	6D	9.06	22.54	16.53	33.08	24.55	3.17
IK-GAT (ours)	6D	7.43	18.60	12.80	24.47	18.08	0.37
Improvement	-	1.63 ▼	3.94 ▼	3.73 ▼	8.61 ▼	6.47 ▼	~8.6x ▼

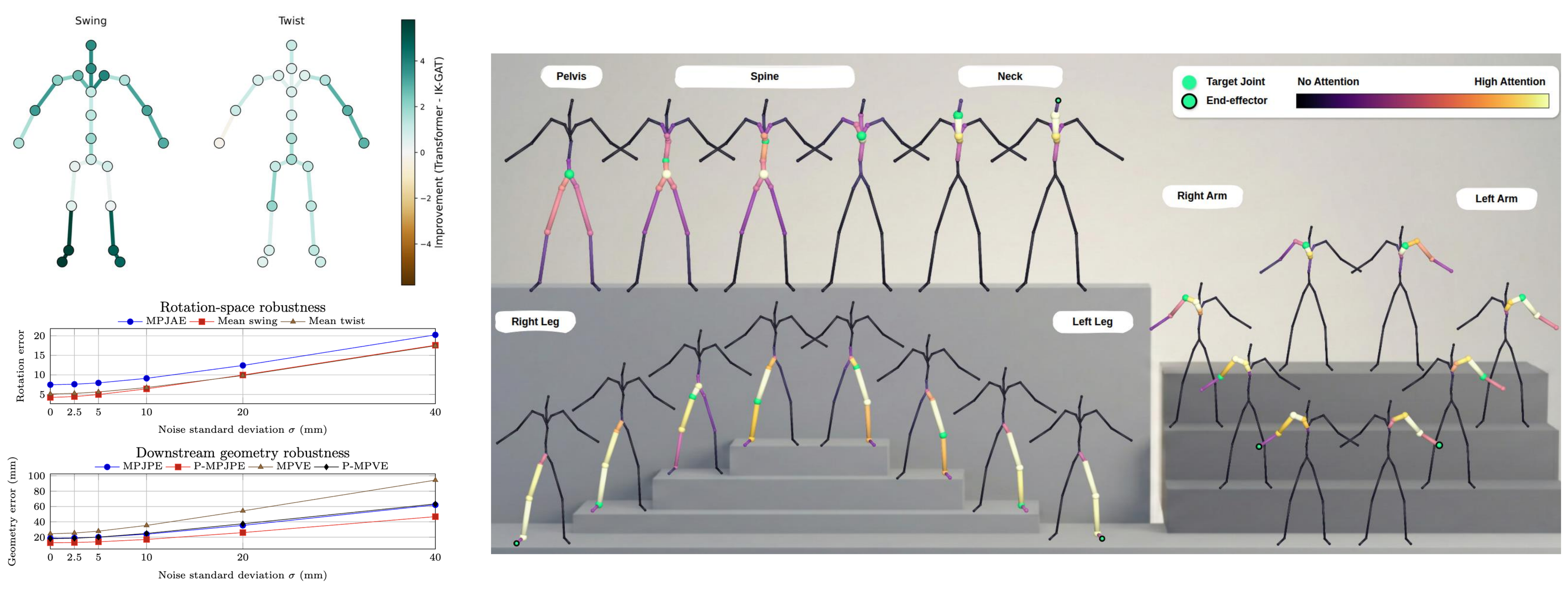
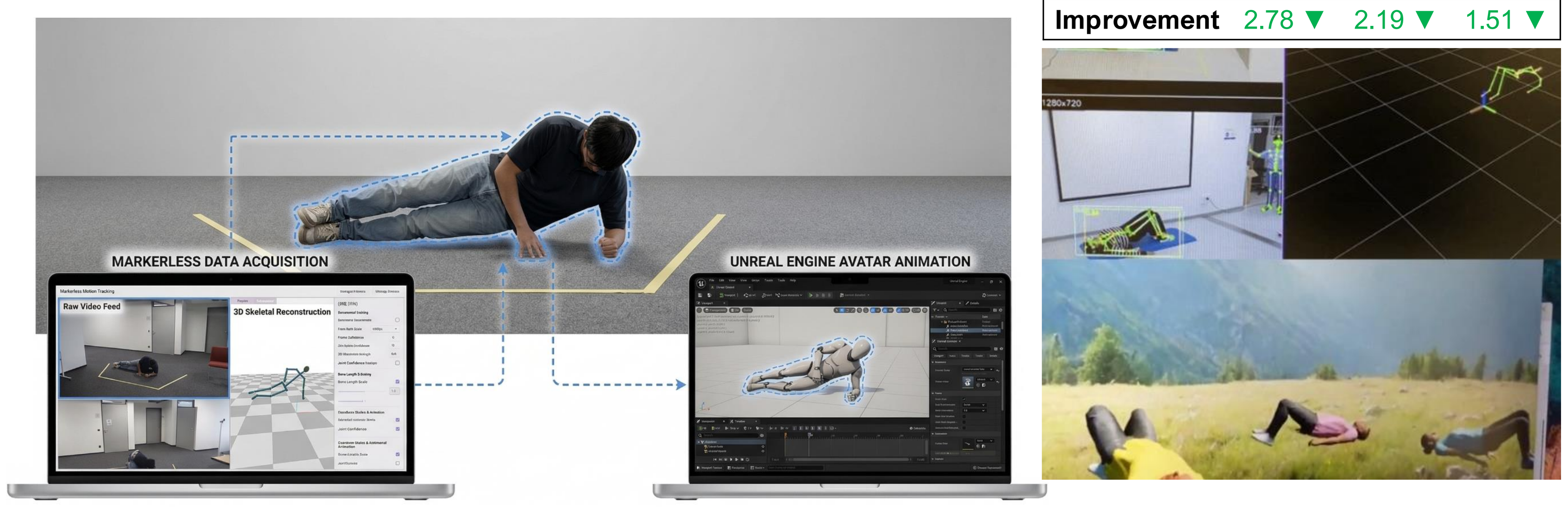


Fig. 6: ***SMPL benchmark and analysis.*** *IK-GAT outperforms a dense transformer using 8.6x fewer parameters, with improvements mainly concentrated in distal bones (left-top), and remains robust under minor noise (left-bottom). Model behavior can be explained by visualizing the graph attention (right), showing that IK-GAT learns to look at related bones when predicting a given joint.*

4(b). Results: Unreal Engine Benchmark

- IK-GAT reduces local-space MPJAE from 15.61° to 12.83° and directly drives production avatars.
- Position-to-rotation at ~650 FPS on CPU.

Method	MPJAE	Swing	Twist
MLP	75.08	38.27	57.79
GCN	30.33	22.01	19.29
ST-CNN	20.64	12.15	14.25
Transformer	15.61	7.39	11.93
IK-GAT (ours)	12.83	5.20	10.42
Improvement	2.78 ▼	2.19 ▼	1.51 ▼



6. Conclusions

- We demonstrate that it is possible to train a lightweight neural network for the inverse orientation recovery task (or neural inverse kinematics) using:
  - graph-based kinematic priors, and
  - more interpretable rotation frames.
- We use our proposed method to demonstrate end-to-end applications with SMPL and rigged game avatar animation.

References

[1] Zhou, Y., Barnes, C., Lu, J., Yang, J., & Li, H. (2019). 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).
[2] Song, X., Li, Z., Chen, S., & Demachi, K. (2024). Quater-gcn: Enhancing 3d human pose estimation with orientation and semi-supervised training. arXiv preprint arXiv:2404.19279.
[3] Mahmood, N., Ghorbani, N., Troje, N. F., Pons-Moll, G., & Black, M. J. (2019). AMASS: Archive of motion capture as surface shapes. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 5442-5451).
[4] Khan, M. S. U., & Wang, C.-Y. (2026). IK-GAT Model for Neural Inverse Kinematics and Unreal Engine IK (UE-IK) Dataset. Zenodo. https://doi.org/10.5281/zenodo.20801456

Contact

Muhammad **Saif** Ullah Khan
Human Motion Researcher
DFKI GmbH
https://saifkhichi.com/