

# AvatarPoser: Articulated Full-Body Pose Tracking from Sparse Motion Sensing

Jiayi Jiang<sup>1</sup>, Paul Streli<sup>1</sup>, Huajian Qiu<sup>1</sup>, Andreas Fender<sup>1</sup>,  
Larissa Laich<sup>2</sup>, Patrick Snape<sup>2</sup>, Christian Holz<sup>1</sup>

<sup>1</sup> Department of Computer Science, ETH Zurich, Switzerland

<sup>2</sup> Reality Labs at Meta, Switzerland

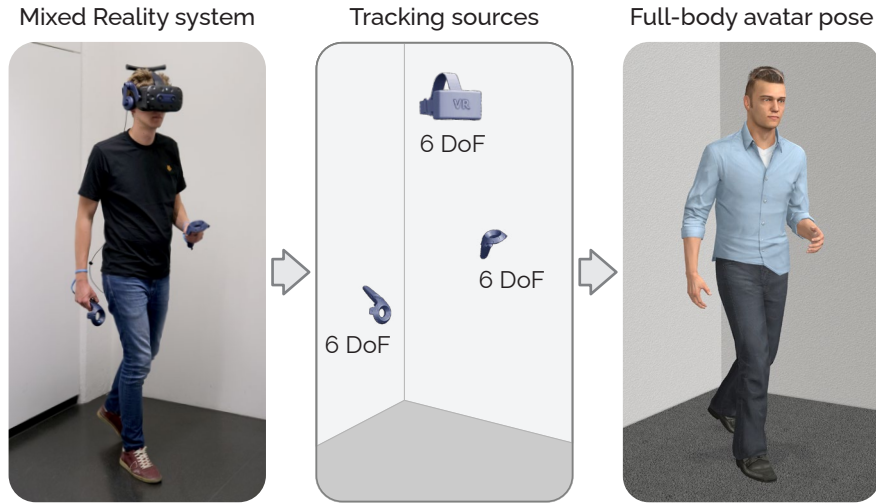
<https://github.com/eth-siplab/AvatarPoser>

**Abstract.** Today’s Mixed Reality head-mounted displays track the user’s head pose in world space as well as the user’s hands for interaction in both Augmented Reality and Virtual Reality scenarios. While this is adequate to support user input, it unfortunately limits users’ virtual representations to just their upper bodies. Current systems thus resort to floating avatars, whose limitation is particularly evident in collaborative settings. To estimate full-body poses from the sparse input sources, prior work has incorporated additional trackers and sensors at the pelvis or lower body, which increases setup complexity and limits practical application in mobile settings. In this paper, we present *AvatarPoser*, the first learning-based method that predicts full-body poses in world coordinates using only motion input from the user’s head and hands. Our method builds on a Transformer encoder to extract deep features from the input signals and decouples global motion from the learned local joint orientations to guide pose estimation. To obtain accurate full-body motions that resemble motion capture animations, we refine the arm joints’ positions using an optimization routine with inverse kinematics to match the original tracking input. In our evaluation, *AvatarPoser* achieved new state-of-the-art results in evaluations on large motion capture datasets (AMASS). At the same time, our method’s inference speed supports real-time operation, providing a practical interface to support holistic avatar control and representation for Metaverse applications.

**Keywords:** 3D Human Pose Estimation, Inverse Kinematics, Augmented Reality, Virtual Reality

## 1 Introduction

Interaction in today’s Mixed Reality (MR) environments is driven by the user’s head pose and input from the hands. Cameras embedded in head-mounted displays (HMD) track the user’s position inside the world and estimate articulated hand poses during interaction, which finds frequent application in Augmented Reality (AR) scenarios. Virtual Reality (VR) systems commonly equip the user with two hand-held controllers for spatial input to render haptic feedback. In



**Fig. 1.** We address the new problem of full-body avatar pose estimation from sparse tracking sources, which can significantly enhance embodiment, presence, and immersion in Mixed Reality. Our novel Transformer-based method *AvatarPoser* takes as input only the positions and orientations of one headset and two handheld controllers (or hands), and generates a full-body avatar pose over 22 joints. Our method reaches state-of-the-art pose accuracy, while providing a practical interface into the Metaverse.

both cases, even this sparse amount of tracking information suffices for interacting with a large variety of immersive first-person experiences.

However, the lack of complete body tracking can break immersion and reduce the fidelity of the overall experience as soon as interactions exceed manual first-person tasks. This not just becomes evident as users see their own bodies during interaction in VR, but also in collaborative tasks in AR that necessarily limit the representation of other participants to their upper bodies, rendered to hover through space. Studies on avatar appearances have shown the importance of holistic avatar representations to achieve embodiment [49] and to establish presence in the virtual environment [19]. Applications such as telepresence or productivity meetings would greatly benefit from more holistic avatar representations that approach the fidelity of motion-capture animations.

This challenge will likely not be addressed by future hardware improvements, as MR systems increasingly optimize for mobile use outside controlled spaces that could accommodate comprehensive tracking. Therefore, we cannot expect future systems to expand much on the tracking information that is available today. While the headset’s cameras may partially capture the user’s feet in opportune moments with a wide field of view, head-mounted cameras are generally in a challenging location for capturing ego-centric poses [50].

Animating a complex full-body avatar based on the sparse input available on today’s platforms is a vastly underdetermined problem. To estimate the com-

plete set of joint positions from the limited tracking sources, previous work has constrained the extend of motion diversity [3] or used additional trackers on the user’s body, such as a 6D pelvis tracker [53] or several body-worn inertial sensors [20]. Dittadi et al.’s recent method estimates full-body poses from only the head and hand poses with promising results [13]. However, since the method encodes all joints relative to the pelvis, it implicitly assumes knowledge of a fourth 3D input (i.e., the pelvis).

For practical application, existing methods for full-body avatar tracking come with three limitations: (1) Most general-purpose applications use Inverse Kinematics (IK) to estimate full-body poses. This often generates human motion that appears static and unnatural, especially for those joints that are far away from the known joint locations in the kinematic chain. (2) Despite the goal of using input from only the head and hands, existing deep learning-based methods implicitly assume knowledge of the pelvis pose. However, pelvis tracking may never be available in most portable MR systems, which increases the difficulty of full-body estimation. (3) Even with a tracked pelvis joint, animations from estimated lower-body joints sometimes contain jitter and sliding artifacts. These tend to arise from unintended movement of the pelvis tracker, which is attached to the abdomen and thus moves differently from the actual pelvis joint.

In this paper, we propose a novel Transformer-based method for full human pose estimation with only the sparse tracking information from the head and hand (or controller) poses as input. With *AvatarPoser*, we decouple the global motion from learned pose features and use it to guide our pose estimation. This provides robust results in the absence of other inputs, such as pelvis location or inertial trackers. To the best of our knowledge, our method is the first to recover the full-body motion from only the three inputs across a wide variety of motion classes. Because the predicted end effector poses of an avatar accumulate errors through the kinematic chain, we optimize our initial parameter estimations through inverse kinematics. This combination of our learning-based method with traditional model-based optimization strikes a good balance between full-body style realism and accurate hand control.

We demonstrate the effectiveness of *AvatarPoser* on the challenging AMASS dataset. Our proposed method achieves state-of-the-art accuracy on full-body avatar estimation from sparse inputs. For inference, our network reaches rates of up to 662 fps. In addition, we test our method on data we recorded with an HTC VIVE system and find good generalization of *AvatarPoser* to unseen user input. Taken together, our method provides a suitable solution for practical applications that operate based on the available tracking information on current MR headsets for application in both, Augmented Reality scenarios and Virtual Reality environments.

## 2 Related Work

**Full-Body Pose Estimation from Sparse Inputs.** Much prior work on full-body pose estimation from sparse inputs has used up to 6 body-worn inertial

sensors [48; 20; 55; 54]. Because these 6 IMUs are distributed over head, arms, pelvis and legs, motion capture becomes inflexible and unwieldy. CoolMoves [3] was first to use input from only the headset and hand-held controllers to estimate full-body poses. However, the proposed KNN-based method interpolates poses from a smaller dataset with only specific motion activities and it is unclear how well it scales to large datasets with diverse subjects and activities, also for inference. LoBSTR [53] used a GRU network to predict the lower-body pose from the past sequence of tracking signals of the head, hands, and pelvis, while it computes the upper-body pose to match the tracked end-effector transformations via an IK solver. The authors also highlight the difficulty of developing a system for estimations from 3 sources only, especially when distinguishing a wide range of human poses due to the large amount of ambiguity. More recently, Dittadi et al. proposed a VAE-based method to generate plausible and diverse body poses from sparse input [13]. However, their method implicitly uses knowledge of the pelvis as a fourth input location by encoding all joints relative to the pelvis, which leaves the highly ill-posed problem with only three inputs unsolved.

**Vision Transformer.** Transformers have achieved great success in their initial application in natural language processing [46; 12; 11]. The use of Transformer-based models has also significantly improved the performance on various computer vision tasks such as image classification [14; 28; 16], image restoration [26; 56; 52], object detection [8; 62; 44], and object tracking [33; 59; 43]. In the area of human pose estimation, METRO [27] was first to apply Transformer models to vertex-vertex and vertex-joint interactions for 3D human pose and mesh reconstruction from a single image. PoseFormer [60] and ST-Transformer [4] used Transformers to capture both body joint correlations and temporal dependencies. MHFormer [25] leveraged the spatio-temporal representations of multiple pose hypotheses to predict 3D human pose from monocular videos. In contrast to their offline setting where the complete time series of motions are available, our method focuses on the practical scenario where streaming data is processed by our Transformer in real-time without looking ahead.

**Inverse Kinematics.** Inverse kinematics (IK) is the process of calculating the variable joint parameters to produce a desired end-effector location. IK has been extensively studied in the past, with various applications in robotics [17; 37; 51; 40; 32] and computer animation [58; 18; 42; 36; 2]. Because no analytical solution usually exists for an IK problem, the most common way to solve the problem is through numerical methods via iterative optimization, which is costly. To speed up computation, several heuristic methods have been proposed to approximate the solution [6; 30; 39; 9]. Recently, learning-based IK solutions have attracted attention [10; 7; 47; 38; 5; 15], because they can speed up inference. However, these methods are usually restricted to a scenario with a known data distribution and may not generalize well. To overcome this problem, recent works have combined IK with deep learning to make the prediction more robust and flexible [41; 24; 53; 21; 23; 57]. Our proposed method combines a deep neural network

with IK optimization, where the IK component of our method refines the arm articulation to match the tracked hand positions from the original input (i.e., position of the hands or hand-held controllers).

### 3 Method

#### 3.1 Problem Formulation

Although MR systems differ in the tracking technology they rely on, the global positions in Cartesian coordinates  $\mathbf{p}^{1 \times 3}$  and orientations in axis-angle representation  $\Phi^{1 \times 3}$  of the headset and the hand-held controllers or hands are generally available. From these, *AvatarPoser* reconstructs the position of the articulated joints of the user’s full body within the world  $\mathbf{w}$ . This mapping  $f$  is described through the following equation,

$$\mathbf{T}_{1:\tau}^{1:F} = f(\{\mathbf{p}^{\mathbf{w}}, \Phi^{\mathbf{w}}\}_{1:\tau}^{1:S}), \quad (1)$$

where  $S$  corresponds to the number of joints tracked by the MR system,  $F$  is the number of joints of the full-body skeleton,  $\tau$  matches the number of observed MR frames that are considered from the past, and  $\mathbf{T} \in SE(3)$  is the body joint pose which is represented by  $\mathbf{T} = \{\mathbf{p}, \Phi\}$ .

Specifically, we use the SMPL model [29] to represent and animate our human body pose. We use the first 22 joints defined in the kinematic tree of the SMPL human skeleton and ignore the pose of fingers similar to previous work [13].

#### 3.2 Input and Output Representation

Since the 6D representation of rotations has proved effective for training neural networks due to its continuity [61], we convert the default axis-angle representation  $\Phi^{1 \times 3}$  in the SMPL model to the rotation matrix  $\mathbf{R}^{3 \times 3}$  and discard the last row to get the 6D rotation representation  $\theta^{1 \times 6}$ . During development, we observed that this 6D representation produces smooth and robust rotation predictions.

In addition to the accessible positions  $\mathbf{p}^{1 \times 3}$  and orientations  $\mathbf{R}^{3 \times 3}$  of the headset and hands, we also calculate the corresponding linear and angular velocities to obtain a signal of temporal smoothness. The linear velocity  $\mathbf{v}$  is given by backward finite difference at each time step  $t$ :

$$\mathbf{v}_t = \mathbf{p}_t - \mathbf{p}_{t-1} \quad (2)$$

Similar, the angular velocity  $\Omega$  can be calculated by:

$$\Omega_t = \mathbf{R}_{t-1}^{-1} \mathbf{R}_t \quad (3)$$

followed by also converting to its 6D representation  $\omega^{1 \times 6}$ . As a result, the final input representation is a concatenated vector of position, linear velocity, rotation, and angular velocity from all given sparse inputs, which we write as:

$$\mathbf{X}_t^{1 \times 18S} = [\{\mathbf{p}_t^1, \mathbf{v}_t^1, \theta_t^1, \omega_t^1\}^{1 \times 18}, \dots, \{\mathbf{p}_t^S, \mathbf{v}_t^S, \theta_t^S, \omega_t^S\}^{1 \times 18}] \quad (4)$$

Therefore, when the number of sparse trackers  $S$  equals 3, the number of input features at each time step is 54.

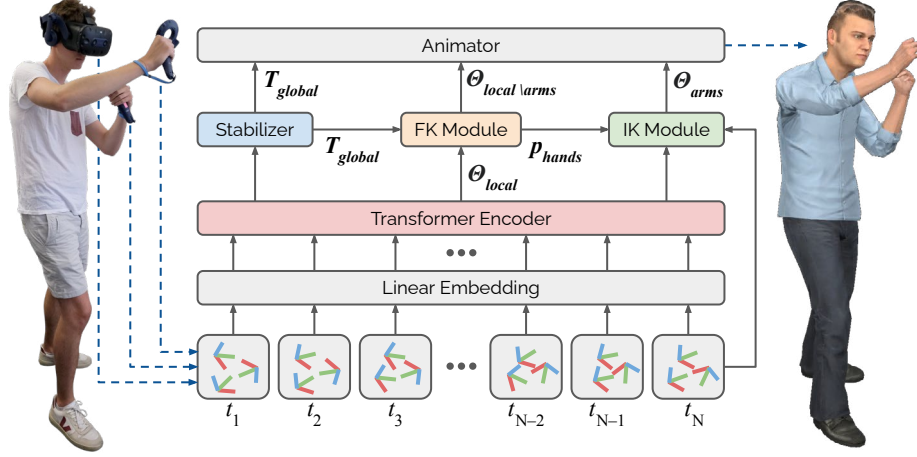
The output of our rotation-based pose estimation network is the local rotation at each joint with respect to the parent joints  $\theta_{\text{local}}$ . The rotation value at the pelvis, which is the root of the SMPL model, refers to the global orientation  $\theta_{\text{global}}$ . As we use 22 joints to represent the full-body motion, the output dimension at each time step is 132.

### 3.3 Overall Framework for Avatar Full-Body Pose Estimation

Fig. 2 illustrates the overall framework of our proposed method *AvatarPoser*. *AvatarPoser* is a time series network that takes as input the 6D signals from the sparse trackers over the previous  $N - 1$  frames and the current  $N^{\text{th}}$  frame and predicts global orientation of the human body as well as the local rotations at each joint with respect to its parent joint. Specifically, *AvatarPoser* consists of four components: a Transformer Encoder, a Stabilizer, a Forward-Kinematics (FK) Module, and an Forward-Kinematics (IK) Module. We designed the network such that each component solves a specific task.

**Transformer Encoder.** Our method builds on a Transformer model to extract the useful information from time-series data, following its benefits in efficiency, scalability, and long-term modeling capabilities. We particularly leverage the Transformer’s self-attention mechanism to distinctly capture global long-range dependencies in the data. Specifically, given the input signals, we apply a linear embedding to enrich the features to 256 dimensions. Next, our Transformer Encoder extracts deep pose features from previous time steps from the headset and hands, which are shared by the Stabilizer for global motion prediction, and a 2-layer multi-layer perceptron (MLP) for local pose estimation, respectively. We set the number of heads to 8 and the number of self-attention layers to 3.

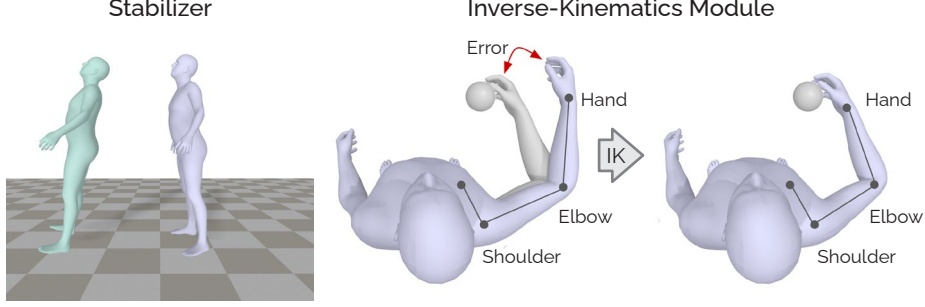
**Stabilizer.** The Stabilizer is a 2-layer MLP that takes as input the 256-dimensional pose features from our Transformer Encoder. We set the number of nodes in the hidden layer to 256. The output of the network produces the estimated global orientation represented as the rotation of the pelvis; therefore, it is responsible for global motion navigation by decoupling global orientation from pose features and obtaining global translation from the head position through the body kinematic chain. Although it may be intuitive and possible to calculate the global orientation from a given head pose through the kinematic chain, the user’s head rotation is often independent of the motions of other joints. As a result, the global orientation at the pelvis is sensitive to the rotation of the head. Considering the scenario where a user stands still and only rotates their head, it is likely that the global orientation may have a large error, which often results in a floating avatar.



**Fig. 2.** The framework of our proposed *AvatarPoser* for Mixed Reality avatar full-pose estimation integrates four parts: a Transformer Encoder, a Stabilizer, a Forward-Kinematics Module, and an Inverse-Kinematics Module. The Transformer Encoder extracts deep pose features from previous time step signals from the headset and hands, which are split into global and local branches and correspond to global and local pose estimation, respectively. The Stabilizer is responsible for global motion navigation by decoupling global orientation from pose features and estimating global translation from the head position through the body’s kinematic chain. The Forward-Kinematics Module calculates joint positions from a human skeleton model and a predicted body pose. The Inverse-Kinematics Module adjusts the estimated rotation angles of joints on the shoulder and elbow to reduce hand position errors.

**Forward-Kinematics Module.** The Forward-Kinematics (FK) Module calculates all joint positions given a human skeleton model and predicted local rotations as input. While rotation-based methods provide robust results without the need to reproject onto skeleton constraints to avoid bone stretching and invalid configurations, they are prone to error accumulating along the kinematic chain. Training the network without FK could only minimize the rotation angles, but would not consider the actually resulting joint positions during optimization.

**Inverse-Kinematics Module.** A main problem of rotation-based pose estimation is that the prediction of end-effectors may deviate from their actual location—even if the end effector served as a known input, such as in the case of hands. This is because for end-effectors, the error accumulates along the kinematic chain. Accurately estimating the position of end-effectors is particularly important in MR, however, because hands typically often used for providing input and even small errors in position can significantly disturb interaction with virtual interface elements. To account for this, we integrate a separate IK algorithm that adjusts the arm limb positions according to the known hand positions.



**Fig. 3. Left:** Our Stabilizer predicts global orientation and, thus, global motion (right avatar) that is significantly more robust than simply aligning the head of predicted body with the known input orientation (left avatar). **Right:** To account for accumulated errors along the joint hierarchy, we integrate an additional IK step to optimize the end-effectors’ locations and match their target positions.

Our methods performs IK-based optimization based on the estimated parameters output by our neural network. This combines the individual benefits of both approaches as explored in prior work (e.g., [24; 53]). Specifically, after our network produces an output, our IK Module adjusts the estimated rotation angles of joints on the shoulder and elbow to reduce the error of hand positions as shown in Fig. 3. We thereby fix the position of the shoulder and do not optimize the other rotation angles, because we found the resulting overall body posture to appear more accurate than the output of the IK algorithm.

Given the initial rotation values  $\theta_0 = \{\theta_0^{\text{shoulder}}, \theta_0^{\text{elbow}}\}$  estimated from our Transformer network, we calculate the positional error  $\mathbf{E}$  of the hand according to the input signals and estimated hand position through the FK Module by

$$\mathbf{E}(\theta_i) = \|\mathbf{p}_{\text{gt}}^{\text{hand}} - \text{FK}(\theta_i)^{\text{hand}}\|_2^2 \quad (5)$$

Then the rotation value is updated by:

$$\theta_{i+1} = \theta_i + \eta \cdot f(\nabla \mathbf{E}(\theta_i)) \quad (6)$$

where  $\eta$  is the learning rate and  $f(\cdot)$  is decided by the specific optimizer. To enable fast inference for real application, we stop the optimization after a fixed number of iterations.

There are several classical non-linear optimization algorithms that are suitable for optimizing inverse kinematics problems, such as Gauss-Newton method or the Levenberg-Marquardt method [34]. In our experiment, we leverage the Adam optimizer [22] due to its compatibility with Pytorch. We set the learning rate as  $1 \times 10^{-3}$ .

**Loss Function.** The final loss function is composed of an L1 local rotational loss, an L1 global orientation loss, and an L1 positional loss, denoted by:

$$\mathbf{L}_{\text{total}} = \lambda_{\text{ori}} \mathbf{L}_{\text{ori}} + \lambda_{\text{rot}} \mathbf{L}_{\text{rot}} + \lambda_{\text{fk}} \mathbf{L}_{\text{fk}} \quad (7)$$



**Table 1.** Comparisons of MPJRE [°], MPJPE [cm], and MPJVE [cm/s] to State-of-the-Arts on AMASS dataset. For each metric, the best result is highlighted in **boldface**.

Methods	Four Inputs			Three Inputs		
	MPJRE	MPJPE	MPJVE	MPJRE	MPJPE	MPJVE
Final IK	12.39	9.54	36.73	16.77	18.09	59.24
CoolMoves	4.58	5.55	65.28	5.20	7.83	100.54
LoBStr	8.09	5.56	30.12	10.69	9.02	44.97
VAE-HMD	3.12	3.51	28.23	4.11	6.83	37.99
AvatarPoser (Ours)	<b>2.59</b>	<b>2.61</b>	<b>22.16</b>	<b>3.21</b>	<b>4.18</b>	<b>29.40</b>

We set the weights  $\lambda_{ori}$ ,  $\lambda_{rot}$ , and  $\lambda_{fk}$  to 0.05, 1, and 1, respectively. For fast training, we do not include our IK Module into the training stage.

## 4 Experiments

### 4.1 Data Preparation and Network Training

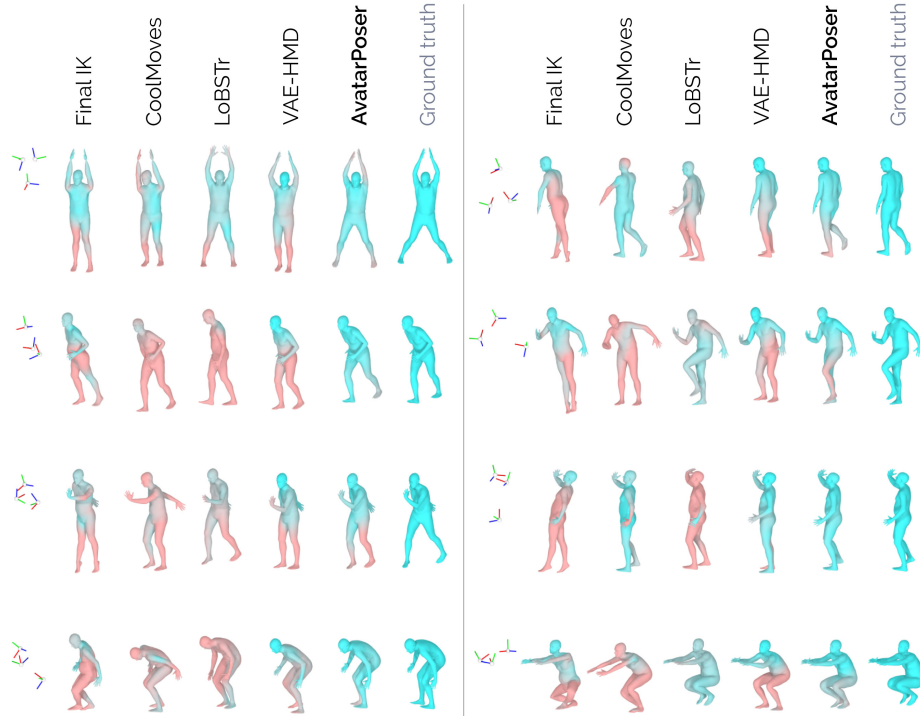
We use the subsets CMU [1], BMLrub [45] and HDM05 [35] in AMASS [31] dataset for training and testing. The AMASS dataset is a large human motion database that unifies different existing optical marker-based MoCap datasets by converting them into realistic 3D human meshes represented by SMPL [29] model parameters. We split the three datasets into random training and test sets with 90% and 10% of the data, respectively. For use on VR devices, we unified the frame rate to 60 Hz.

To optimize the parameters of *AvatarPoser*, we adopt the Adam solver [22] with batch size 256. We set the chunk size of input as 40 frames. The learning rate starts from  $1 \times 10^{-4}$  and decays by a factor of 0.5 every  $2 \times 10^4$  iterations. We train our model with PyTorch on one NVIDIA GeForce GTX 3090 GPU. It takes about two hours to train *AvatarPoser*.

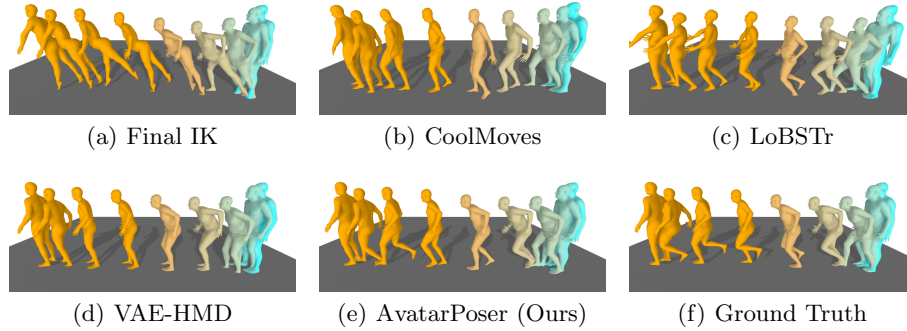
### 4.2 Evaluation Results

We use MPJRE (Mean Per Joint Rotation Error [°]), MPJPE (Mean Per Joint Position Error [cm]), and MPJVE (Mean Per Joint Velocity Error [cm/s]) as our evaluation metrics. We compare our proposed *AvatarPoser* with Final IK [2], CoolMoves [3], LoBStr [53], and VAE-HMD [13], which are state-of-the-art methods working on the problems of avatar pose estimation from sparse inputs.

Since these state-of-the-art methods do not provide public source codes, we directly run Final IK in Unity [2] and reproduce other methods to the best of our knowledge. For a fair comparison, we train all the methods on the same training and testing data. It should be noted that the original CoolMoves is a position-based method, we adapt it to rotation-based method for a fair comparison with other methods. We make all the methods work with both three (headset, controllers) and four inputs (headset, controllers, pelvis tracker). When only three



**Fig. 4.** Visual comparisons of different methods based on given sparse inputs for various motions. Avatars are color-coded to show errors in red.



**Fig. 5.** Visual results of our proposed method *AvatarPoser* compared to SOTA alternatives for a running motion. The change of color denotes different timestamp.

inputs are provided, for input representation we do not use the pose of pelvis as a reference frame, and for the output we calculate the global orientation and translation of human body at pelvis through the kinematic chains from the given global pose of head.

The numerical results for the considered metrics (MRJRE, MPJPE, and MPJVE) for both four and three inputs are reported in Table 1. It can be seen that our proposed *AvatarPoser* achieves the best results on all three metrics and outperforms all other methods. VAE-HMD achieves the second best performance on MPJPE, which is followed by CoolMoves (KNN). Final IK gives the worst result on MPJPE and MPJRE because it optimizes the pose of the end-effectors without considering the smoothness of other body joints. As a result, the performance of LoBSTR, which uses Final IK for upper body pose estimation, is also low. We believe this shows the value in data-driven methods to learn motion from existing mocap datasets. However, it does not mean that traditional optimization methods are not useful. In our ablation studies, we show how inverse kinematics when combined with deep learning can improve the accuracy of hand positions.

To further evaluate the generalization ability of our proposed method, we perform a 3-fold cross-dataset evaluation among different methods. To do so, we train on two subsets and test on the other subset in a round robin fashion. Table 2 shows the experimental results of different methods tested on CMU, BMLrub, and HDM05 datasets. We achieve the best results over almost all evaluation metrics in all three datasets. Although Final IK performs slightly better than *AvatarPoser* in terms of MPJVE in CMU, which can only mean the motions are a little bit smoother. However, the rotation error MPJRE and the position error MPJPE of Final IK, which represent the accuracy of predictions, are much larger than our method.

**Table 2.** Results of cross-dataset evaluation between different methods. The input signals are from only three devices, i.e., one headset and two controllers. The best results for each dataset and each evaluation metrics are highlighted in **boldface**.

Dataset	Methods	MPJRE	MPJPE	MPJVE
CMU	Final IK	17.80	18.82	56.83
	CoolMoves	9.20	18.77	139.17
	LoBSTR	12.51	12.96	49.94
	VAE-HMD	6.53	13.04	51.69
	AvatarPoser (Ours)	<b>5.93</b>	<b>8.37</b>	<b>35.76</b>
BMLrub	Final IK	15.93	17.58	60.64
	CoolMoves	7.93	13.30	134.77
	LoBSTR	10.79	11.00	60.74
	VAE-HMD	5.34	9.69	51.80
	AvatarPoser (Ours)	<b>4.92</b>	<b>7.04</b>	<b>43.70</b>
HDM05	Final IK	18.64	18.43	62.39
	CoolMoves	9.47	17.90	140.61
	LoBSTR	13.17	11.94	48.26
	VAE-HMD	6.45	10.21	40.07
	AvatarPoser (Ours)	<b>6.39</b>	<b>8.05</b>	<b>30.85</b>

**Table 3.** Ablation studies. Best results are highlighted in **bold** for each metric.

Configurations	MPJRE	MPJPE-Full Body	MPJPE-Hand
Default	6.39	<b>8.05</b>	<b>1.86</b>
No Stabilizer	6.39	9.29	2.15
Predict Pelvis Position	6.42	8.82	2.11
No FK Module	<b>6.24</b>	8.41	2.04
No IK Module	6.41	8.07	3.17

### 4.3 Ablation Studies

We perform an ablation study on the different submodules of our method and provide results in Table 3. The experiments are conducted on the same test set as HDM05 in Table 2. We use MPJRE [°], MPJPE [cm] as our evaluation metrics in the ablation studies to show the need for each component. In addition to the position error across the full-body joints, we specifically calculate the mean error on hands to show how the IK module helps improve the hand positions.

**No Stabilizer.** We remove the Stabilizer module, which predicts the global orientation, and calculate the global orientation through the body kinematic chain directly from the given orientation of the head. Table 3 shows that the MPJPE drops without Stabilizer. This is because the rotation of the head is relatively independent to the rest of the body. Therefore, the global orientation is highly sensitive to random rotations of the head. Learning the global orientation from richer information via the network is a superior way to solve the problem.

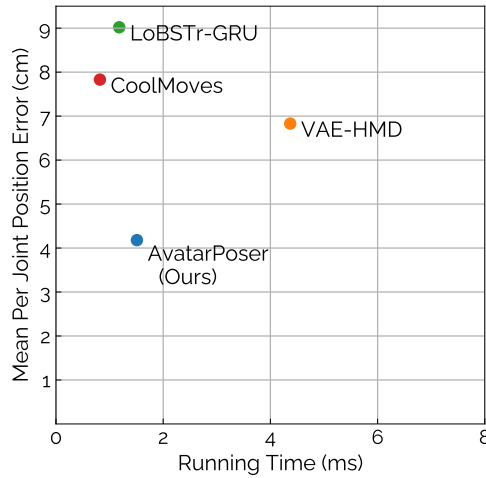
**Predict Pelvis Position.** In our final model, we calculate the global translation of the human body, which is located at the pelvis, from the input head position through the kinematic chain. We also try directly regressing to the global translation within the network, but the result is worse than computing via the kinematic chain according to our evaluation results.

**No FK Module.** We also remove the FK Module, which means the network is only trained to minimize the rotation angles without considering the positions of joints after forward kinematics calculation. When we remove the FK module, the MPJPE increases and the MPJRE decreases. This is intuitive as we only optimize the joint rotations without the IK module. While rotation-based methods provide robust results without the need to reproject onto skeleton constraints to avoid bone stretching and invalid configurations, they are prone to error accumulation along the kinematic chain.

**No IK Module.** We remove IK Module and only provide the results directly predicted by our neural network. Removing the IK module has little effect on the average position error of full-body joints. However, the average position error of the hands increases by almost 41%.

#### 4.4 Running Time Analysis

We evaluated the run-time inference performance of our network *AvatarPoser* and compared it to the inference of VAE-HMD [13], LoBSTR [53], CoolMoves [3] as shown in Fig. 6. Note that we did not include Final IK [2], because its integration into Unity makes accurate measurements difficult. To conduct our comparison, we modified LoBSTR to directly predict full-body motion via the GRU (denoted as LoBSTR-GRU) instead of combining Final IK and the GRU together. We measured the run time per frame (in milliseconds) on the evaluated test set on one NVIDIA 3090 GPU. For a fair comparison, we only calculated the network inference time of *AvatarPoser* here. Our *AvatarPoser* achieves a good trade-off between performance and inference speed.



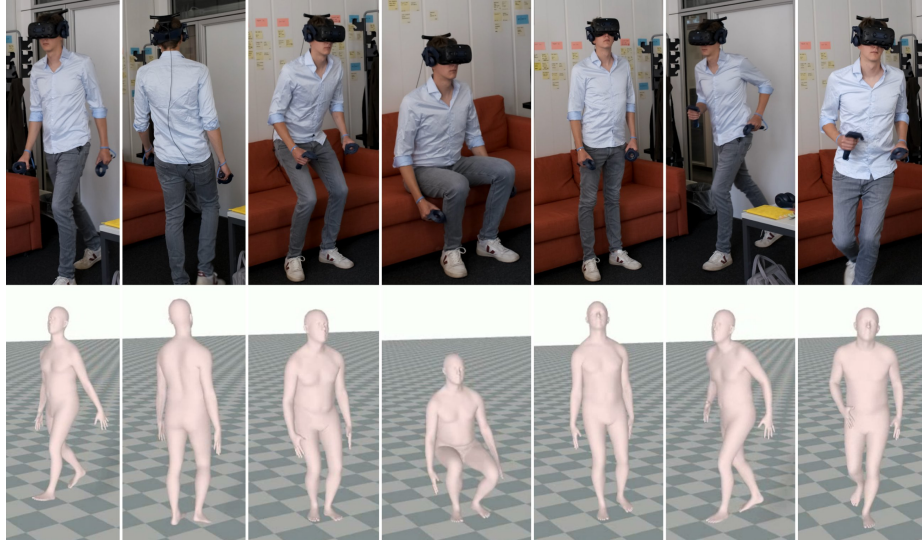
**Fig. 6.** Comparisons of network inference time across several methods. Due to the powerful and efficient Transformer encoder, our method achieves the smallest average position error while providing fast inference.

Our method also requires executing an IK algorithm after the network forward pass. Each iteration costs approximately 6 ms, so we set the number of iterations to 5 to keep a balance between inference speed and the accuracy of the final hand position. Note that the speed could be accelerated by adopting a more standard non-linear optimization.

#### 4.5 Test on a Commercial VR System

To qualitatively assess the robustness of our method, we executed our algorithm on live recordings from an actual VR system. We used an HTC VIVE HMD as well as two controllers, each providing real-time input with six degrees of

freedom (rotation and translation). Fig. 7 shows a few examples of our method’s output based on sparse inputs.



**Fig. 7.** We tested our method on recorded motion data from a VIVE Pro headset and two VIVE controllers. Columns show the user’s pose (top) and our prediction (bottom).

## 5 Conclusions

We presented our novel Transformer-based method *AvatarPoser* to estimate realistic human poses from just the motion signals of a Mixed Reality headset and the user’s hands or hand-held controllers. By decoupling the global motion information from learned pose features and using it to guide pose estimation, we achieve robust estimation results in the absence of pelvis signals. By combining learning-based methods with traditional model-based optimization, we keep a balance between full-body style realism and accurate hand control. Our extensive experiments on the AMASS dataset demonstrated that *AvatarPoser* surpasses the performance of state-of-the-art methods and, thus, provides a useful learning-based IK solution for practical VR/AR applications.

**Acknowledgments:** We thank Christian Knieling for his early explorations of learning-based methods for pose estimation with us at ETH Zürich. We thank Zhi Li, Xianghui Xie, and Dengxin Dai from Max Planck Institute for Informatics for their helpful discussions. We also thank Olga Sorkine-Hornung and Alexander Sorkine-Hornung for early discussions.

## Bibliography

- [1] CMU MoCap Dataset. <http://mocap.cs.cmu.edu/> (2004) 9
- [2] RootMotion Final IK. <https://assetstore.unity.com/packages/tools/animation/final-ik-14290> (2018) 4, 9, 13
- [3] Ahuja, K., Ofek, E., Gonzalez-Franco, M., Holz, C., Wilson, A.D.: Cool-moves: User motion accentuation in virtual reality. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* **5**(2), 1–23 (2021) 3, 4, 9, 13
- [4] Aksan, E., Kaufmann, M., Cao, P., Hilliges, O.: A spatio-temporal transformer for 3d human motion prediction. *International Conference on 3D Vision (3DV)* (2021) 4
- [5] Ames, B., Morgan, J.: Ikflow: Generating diverse inverse kinematics solutions. *IEEE Robotics and Automation Letters* (2022) 4
- [6] Aristidou, A., Lasenby, J.: Fabrik: A fast, iterative solver for the inverse kinematics problem. *Graphical Models* **73**(5), 243–260 (2011) 4
- [7] Bócsi, B., Nguyen-Tuong, D., Csató, L., Schoelkopf, B., Peters, J.: Learning inverse kinematics with structured prediction. In: *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. pp. 698–703. IEEE (2011) 4
- [8] Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: *European conference on computer vision*. pp. 213–229. Springer (2020) 4
- [9] Çavdar, T., Mohammad, M., Milani, R.A.: A new heuristic approach for inverse kinematics of robot arms. *Advanced Science Letters* **19**(1), 329–333 (2013) 4
- [10] Csiszar, A., Eilers, J., Verl, A.: On solving the inverse kinematics problem using neural networks. In: *2017 24th International Conference on Mechatronics and Machine Vision in Practice (M2VIP)*. pp. 1–6. IEEE (2017) 4
- [11] Dai, Z., Yang, Z., Yang, Y., Carbonell, J.G., Le, Q., Salakhutdinov, R.: Transformer-xl: Attentive language models beyond a fixed-length context. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. pp. 2978–2988 (2019) 4
- [12] Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. *Annual Conference of the North American Chapter of the Association for Computational Linguistics* (2019) 4
- [13] Dittadi, A., Dziadzio, S., Cosker, D., Lundell, B., Cashman, T.J., Shotton, J.: Full-body motion from a single head-mounted device: Generating smpl poses from partial observations. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 11687–11697 (2021) 3, 4, 5, 9, 13



- [14] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2021) 4
- [15] Duka, A.V.: Neural network based inverse kinematics solution for trajectory tracking of a robotic arm. *Procedia Technology* **12**, 20–27 (2014) 4
- [16] Fan, H., Xiong, B., Mangalam, K., Li, Y., Yan, Z., Malik, J., Feichtenhofer, C.: Multiscale vision transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6824–6835 (2021) 4
- [17] Goldenberg, A., Benhabib, B., Fenton, R.: A complete generalized solution to the inverse kinematics of robots. *IEEE Journal on Robotics and Automation* **1**(1), 14–20 (1985) 4
- [18] Grochow, K., Martin, S.L., Hertzmann, A., Popović, Z.: Style-based inverse kinematics. In: ACM SIGGRAPH 2004 Papers, pp. 522–531 (2004) 4
- [19] Heidicker, P., Langbehn, E., Steinicke, F.: Influence of avatar appearance on presence in social vr. In: 2017 IEEE Symposium on 3D User Interfaces (3DUI). pp. 233–234 (2017) 2
- [20] Huang, Y., Kaufmann, M., Aksan, E., Black, M.J., Hilliges, O., Pons-Moll, G.: Deep inertial poser: Learning to reconstruct human pose from sparse inertial measurements in real time. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)* **37**, 185:1–185:15 (Nov 2018) 3, 4
- [21] Kang, M., Cho, Y., Yoon, S.E.: Rcik: Real-time collision-free inverse kinematics using a collision-cost prediction network. *IEEE Robotics and Automation Letters* **7**(1), 610–617 (2021) 4
- [22] Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: International Conference on Learning Representations (2015) 8, 9
- [23] 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) 4
- [24] Li, S., Jiang, J., Ruppel, P., Liang, H., Ma, X., Hendrich, N., Sun, F., Zhang, J.: A mobile robot hand-arm teleoperation system by vision and imu. In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 10900–10906. IEEE (2020) 4, 8
- [25] Li, W., Liu, H., Tang, H., Wang, P., Van Gool, L.: Mhformer: Multi-hypothesis transformer for 3d human pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13147–13156 (2022) 4
- [26] Liang, J., Cao, J., Sun, G., Zhang, K., Van Gool, L., Timofte, R.: Swinir: Image restoration using swin transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1833–1844 (2021) 4
- [27] 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) 4



- [28] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10012–10022 (2021) 4
- [29] Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: Smpl: A skinned multi-person linear model. *ACM transactions on graphics (TOG)* **34**(6), 1–16 (2015) 5, 9
- [30] Luenberger, D.G., Ye, Y., et al.: Linear and nonlinear programming, vol. 2. Springer (1984) 4
- [31] Mahmood, N., Ghorbani, N., Troje, N.F., Pons-Moll, G., Black, M.J.: AMASS: Archive of motion capture as surface shapes. In: International Conference on Computer Vision. pp. 5442–5451 (Oct 2019) 9
- [32] Marić, F., Giamou, M., Hall, A.W., Khoubyarian, S., Petrović, I., Kelly, J.: Riemannian optimization for distance-geometric inverse kinematics. *IEEE Transactions on Robotics* **38**(3), 1703–1722 (2021) 4
- [33] Meinhardt, T., Kirillov, A., Leal-Taixe, L., Feichtenhofer, C.: Trackformer: Multi-object tracking with transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8844–8854 (2022) 4
- [34] Moré, J.J.: The levenberg-marquardt algorithm: implementation and theory. In: Numerical analysis, pp. 105–116. Springer (1978) 8
- [35] Müller, M., Röder, T., Clausen, M., Eberhardt, B., Krüger, B., Weber, A.: Documentation mocap database hdm05. Tech. Rep. CG-2007-2, Universität Bonn (June 2007) 9
- [36] Parger, M., Mueller, J.H., Schmalstieg, D., Steinberger, M.: Human upper-body inverse kinematics for increased embodiment in consumer-grade virtual reality. In: Proceedings of the 24th ACM symposium on virtual reality software and technology. pp. 1–10 (2018) 4
- [37] Parker, J.K., Khoogar, A.R., Goldberg, D.E.: Inverse kinematics of redundant robots using genetic algorithms. In: 1989 IEEE International Conference on Robotics and Automation. pp. 271–272. IEEE Computer Society (1989) 4
- [38] Ren, H., Ben-Tzvi, P.: Learning inverse kinematics and dynamics of a robotic manipulator using generative adversarial networks. *Robotics and Autonomous Systems* **124**, 103386 (2020) 4
- [39] Rokbani, N., Casals, A., Alimi, A.M.: Ik-fa, a new heuristic inverse kinematics solver using firefly algorithm. In: Computational intelligence applications in modeling and control, pp. 369–395. Springer (2015) 4
- [40] Ruppel, P., Hendrich, N., Starke, S., Zhang, J.: Cost functions to specify full-body motion and multi-goal manipulation tasks. In: 2018 IEEE International Conference on Robotics and Automation (ICRA). pp. 3152–3159. IEEE (2018) 4
- [41] Starke, S., Zhang, H., Komura, T., Saito, J.: Neural state machine for character-scene interactions. *ACM Trans. Graph.* **38**(6), 209–1 (2019) 4
- [42] Sumner, R.W., Zwicker, M., Gotsman, C., Popović, J.: Mesh-based inverse kinematics. *ACM transactions on graphics (TOG)* **24**(3), 488–495 (2005) 4

- [43] Sun, P., Cao, J., Jiang, Y., Zhang, R., Xie, E., Yuan, Z., Wang, C., Luo, P.: Transtrack: Multiple object tracking with transformer. arXiv preprint arXiv:2012.15460 (2020) [4](#)
- [44] Sun, Z., Cao, S., Yang, Y., Kitani, K.M.: Rethinking transformer-based set prediction for object detection. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 3611–3620 (2021) [4](#)
- [45] Troje, N.F.: Decomposing biological motion: A framework for analysis and synthesis of human gait patterns. *Journal of vision* **2**(5), 2–2 (2002) [9](#)
- [46] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. *Advances in neural information processing systems* **30** (2017) [4](#)
- [47] Villegas, R., Yang, J., Ceylan, D., Lee, H.: Neural kinematic networks for unsupervised motion retargetting. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 8639–8648 (2018) [4](#)
- [48] Von Marcard, T., Rosenhahn, B., Black, M.J., Pons-Moll, G.: Sparse inertial poser: Automatic 3d human pose estimation from sparse imus. In: *Computer graphics forum*. vol. 36, pp. 349–360. Wiley Online Library (2017) [4](#)
- [49] Waltemate, T., Gall, D., Roth, D., Botsch, M., Latoschik, M.E.: The impact of avatar personalization and immersion on virtual body ownership, presence, and emotional response. *IEEE Transactions on Visualization and Computer Graphics* **24**(4), 1643–1652 (2018) [2](#)
- [50] Wang, J., Liu, L., Xu, W., Sarkar, K., Theobalt, C.: Estimating egocentric 3d human pose in global space. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 11500–11509 (2021) [2](#)
- [51] Wang, L.C., Chen, C.C.: A combined optimization method for solving the inverse kinematics problems of mechanical manipulators. *IEEE Transactions on Robotics and Automation* **7**(4), 489–499 (1991) [4](#)
- [52] Wang, Z., Cun, X., Bao, J., Zhou, W., Liu, J., Li, H.: Uformer: A general u-shaped transformer for image restoration. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 17683–17693 (2022) [4](#)
- [53] Yang, D., Kim, D., Lee, S.H.: Lobstr: Real-time lower-body pose prediction from sparse upper-body tracking signals. In: *Computer Graphics Forum*. vol. 40, pp. 265–275. Wiley Online Library (2021) [3](#), [4](#), [8](#), [9](#), [13](#)
- [54] Yi, X., Zhou, Y., Habermann, M., Shimada, S., Golyanik, V., Theobalt, C., Xu, F.: Physical inertial poser (pip): Physics-aware real-time human motion tracking from sparse inertial sensors. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13167–13178 (2022) [4](#)
- [55] Yi, X., Zhou, Y., Xu, F.: Transpose: real-time 3d human translation and pose estimation with six inertial sensors. *ACM Transactions on Graphics (TOG)* **40**(4), 1–13 (2021) [4](#)
- [56] Zamir, S.W., Arora, A., Khan, S., Hayat, M., Khan, F.S., Yang, M.H.: Restormer: Efficient transformer for high-resolution image restoration. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5728–5739 (2022) [4](#)

- [57] Zhang, X., Bhatnagar, B.L., Guзов, V., Starke, S., Pons-Moll, G.: Couch: Towards controllable human-chair interactions. In: European Conference on Computer Vision (ECCV). Springer (October 2022) [4](#)
- [58] Zhao, J., Badler, N.I.: Inverse kinematics positioning using nonlinear programming for highly articulated figures. *ACM Transactions on Graphics (TOG)* **13**(4), 313–336 (1994) [4](#)
- [59] Zhao, Z., Wu, Z., Zhang, Y., Li, B., Jia, J.: Tracking objects as pixel-wise distributions. In: Proceedings of the European Conference on Computer Vision (ECCV) (2022) [4](#)
- [60] Zheng, C., Zhu, S., Mendieta, M., Yang, T., Chen, C., Ding, Z.: 3d human pose estimation with spatial and temporal transformers. Proceedings of the IEEE International Conference on Computer Vision (ICCV) (2021) [4](#)
- [61] 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) [5](#)
- [62] Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable detr: Deformable transformers for end-to-end object detection. In: International Conference on Learning Representations (2020) [4](#)