

Self-Avatar Animation in Virtual Reality: Impact of Motion Signals Artifacts on the Full-Body Pose Reconstruction

Antoine Maiorca*
ISIA Lab, UMONS, Belgium

Seyed Abolfazl Ghasemzadeh †
ICTEAM/ELEN, UCLouvain, Louvain-la-Neuve, Belgium

Thierry Ravet ‡
ISIA Lab, UMONS, Mons, Belgium

François Cresson §
ISIA Lab, UMONS, Mons, Belgium

Thierry Dutoit ¶
ISIA Lab, University of Mons, Belgium

Christophe De Vleeschouwer ¶
ICTEAM/ELEN, UCLouvain,
Louvain-la-Neuve, Belgium

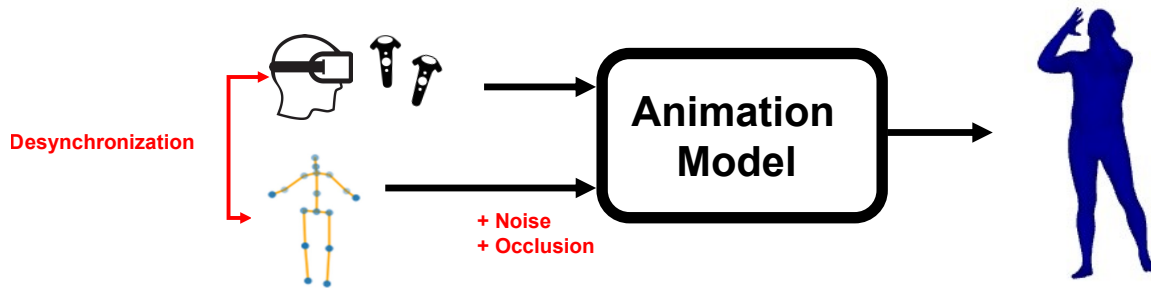


Figure 1: Motion data from sparse inputs are extended with 3D Cartesian positions for leveraging the pose ambiguity in order to reconstruct the full pose of the self-avatar. However, depending on the 3D tracking solution, desynchronization between VR devices motion signals and 3D Cartesian coordinates as well as motion artifacts such as noise and occlusions can arise. With this setup, we can measure the impact of each issue individually in the full-body pose reconstruction.

ABSTRACT

Virtual Reality (VR) applications have revolutionized user experiences by immersing individuals in interactive 3D environments. These environments find applications in numerous fields, including healthcare, education, or architecture. A significant aspect of VR is the inclusion of self-avatars, representing users within the virtual world, which enhances interaction and embodiment. However, generating lifelike full-body self-avatar animations remains challenging, particularly in consumer-grade VR systems, where lower-body tracking is often absent. One method to tackle this problem is by providing an external source of motion information that includes lower body information such as full Cartesian positions estimated from RGB(D) cameras. Nevertheless, the limitations of these systems are multiples: the desynchronization between the two motion sources and occlusions are examples of significant issues that hinder the implementations of such systems. In this paper, we aim to measure the impact on the reconstruction of the articulated self-avatar’s full-body pose of (1) the latency between the VR motion features and estimated positions, (2) the data acquisition rate, (3) occlusions, and (4) the inaccuracy of the position estimation algorithm. In addition, we analyze the motion reconstruction errors using ground truth and 3D Cartesian coordinates estimated from *YOLOv8* pose estimation. These analyzes show that the studied methods are significantly sensitive to any degradation tested, especially regarding the velocity reconstruction error.

*e-mail: antoine.maiorca@umons.ac.be

†e-mail: seyed.ghasemzadeh@uclouvain.be

‡e-mail: thierry.ravet@umons.ac.be

§e-mail: francois.cresson@umons.ac.be

¶e-mail: thierry.dutoit@umons.ac.be

¶e-mail: christophe.devleeschouwer@uclouvain.be

Index Terms: Computing methodologies—Computer graphics—Animation—; Computing methodologies—Computer graphics—Animation—Motion capture

1 INTRODUCTION

Virtual Reality (VR) applications create a 3D environment that immerses users and enables them to interact with virtual elements. The versatility and interactivity of VR technology have led to a wide array of practical uses spanning various industries and sectors. Besides gaming and entertainment, VR finds valuable applications in healthcare [4, 36], education and training [10, 18, 42], as well as architecture and construction engineering [3, 12].

Several studies have highlighted the significance of including a self-avatar, representing the user’s body within the virtual environment. These investigations have demonstrated its favorable effects across multiple domains, including user interaction and embodiment [34], cognitive processes of participants [44], and collaborative tasks within shared virtual spaces [33]. Furthermore, the lifelike movements of the avatar contribute to fostering more realistic and engaging social interactions among users within immersive environments [39]. Hence, one crucial task in the design of VR applications is the animation of the articulated full-body self-avatar.

Typically, VR systems designed for consumers consist of a Head-Mounted Display (HMD) along with optional handheld controllers. The HMD is a wearable device worn on the head and positioned in front of the user’s eyes to present a visual display. These devices incorporate tracking technology to determine their position and orientation. Using this set of motion data (referred as *sparse inputs* in this document), the system generates the user’s complete body movements. However, tracking for the lower body is not included, making it a complex task to synthesize the motion of this body subset. Indeed, synthesizing the full-body motion exclusively from the sparse inputs corresponds to a one-to-many problem since several poses can resolve the motion generation task from one input configuration.

Deep Learning-based animation models have been proposed to tackle this pose ambiguity by learning the complex relationship between the sparse inputs and the lower-body motion [11, 20, 41, 59]. One significant advantage of these methods is their independence from additional tracking devices, beyond the HMD and controllers, to reconstruct the complete self-avatar’s body pose. This feature makes them suitable for consumer-grade applications but the precision in the pose reconstruction can be affected by the lack of available information.

Another approach to mitigate the issue of pose ambiguity involves incorporating external sources of motion data, particularly for the lower-body pose. These additional motion features, that often integrating lower-body limbs information, aim to guide the full-body pose reconstruction [16, 24, 53, 56]. However, these methods are associated with several drawbacks. Firstly, the need for external devices may hinder the widespread adoption of such technologies. Additionally, equipping users with these sensors can potentially affect comfort and detrimentally impact the overall user experience. To avoid this effect, the full-body Cartesian position can be estimated from RGB videos [7, 22] and act as the additional motion information.

In this configuration, the latency between VR motion signals and the Cartesian position is a crucial factor that needs to be taken into account in the implementation of a VR animation system. Hence, the process of integrating, into a unified framework, motion capture data from diverse sources, each operating at its distinct framerate, can be a challenging task. Indeed, each source needs to be synchronized to avoid discrepancies and ensure the coherent representation of the user’s movements. This synchronization is essential because misalignments or desynchronization among the data streams can lead to inaccuracies in the reconstructed body pose, which can significantly impact the overall quality and realism of the animation. Moreover, depending of the additional tracking solution and the number of users to track, it can significantly increase the latency between the two motion sources and have implications for the real-time responsiveness of the animation system, which is a crucial factor in this context.

Finally, the pose estimation from RGB(D) videos suffers from a low-fidelity pose reconstruction compared to physical motion capture devices such as optical markers. Low accuracy on the joint position and artifacts such as occlusion *i.e.*, a hidden information resulting in a missing joint in the estimated pose, are major concerns for animating self-avatar’s based on this set of motion features.

The goal of this paper is to analyze the impact in the articulated self-avatar’s full-body pose reconstruction of:

- the latency between the Cartesian position estimated from RGB videos and the VR motion signals
- the discrepancy between the motion acquisition rate of these two sources of motion
- the motion artifacts that can occur in the estimation of the full-body Cartesian coordinates

To do so, we propose to manually degrades the 3D Cartesian positions with the artifacts described as above. The animation model is then fed by the sparse inputs concatenated with these 3D positions. Finally, we assess the reconstruction error across the tested configurations. Moreover, we suggest extracting the 3D Cartesian coordinates using the pose estimation algorithm embedded in *YOLOv8* [22]. This approach enables us to discern the disparities in model performance when utilizing either ground truth or 3D Cartesian positions estimated in a real-world use case.

2 RELATED WORK

Approaches addressing the animation of a full-body self-avatar in VR can be broadly categorized into two main groups: those relying

solely on sparse inputs and those augmenting VR motion signals with extra motion data. These approaches are respectively explained in Section 2.1 and 2.2.

2.1 Self-avatar’s full-body estimation from sparse VR sensors

Tracking a user’s full-body motion based on sparse sensors is a widely explored topic [13, 23, 24, 57]. In the VR paradigm, the full motion tracking is performed using the motion signals from the VR devices. The self-avatar’s full-body motion is synthesized from the hands and head motion features. Solutions based on Inverse Kinematics (IK) have been implemented to leverage this problem [8, 19, 26, 46].

Then, machine learning techniques have further been used to improve the quality of the full-body motion reconstruction. In the case of *CoolMoves* [1], it leverages k-NN techniques to find a pose that closely corresponds to the sparse inputs within a well-structured motion database. Similarly, *MMVR* [37] adopts motion matching [6] as an alternative approach to achieve real-time animation with smooth transitions. The major drawback of these approaches is the fact that we must ensure that the motion database gathers realistic samples that include not only seamless transitions and smooth blending between distinct motions but also a diverse array of desired actions. Moreover, animating upper body gestures using motion matching is challenging due to the unconstrained nature of user’s arm movements. This complexity necessitates the creation of an extensive and challenging-to-manage motion database to encompass the multitude of feasible poses.

With the emergence of Deep Learning, methods based on artificial neural networks have been designed to capture the complex spatial-temporal dependencies between the sparse signals and the full body pose, especially regarding the low body information where no tracked data are available. More specifically, algorithms built upon architectures designed for time series analysis have been proposed to compute the self-avatar’s full pose based on the sparse inputs [20, 41, 54, 59]. While LSTMs have been successfully employed in this context [54], Transformer-based methods outperformed this solution regarding the quality of the pose reconstruction. *Avatar-Poser* [20] has integrated the encoder part of the transformer architecture [50] to compute high-dimensional embeddings from the sparse information to estimate the avatar’s full pose and its global displacement. *Dual Attention Poser* [59] have proposed to decouple the global and local motion features to feed two transformers and further merge the computed information.

An alternative approach to address this challenge involves generative models. *VAE-HMD* [11] trained a Variational AutoEncoder (VAE) to generate a latent vector from a sequence of sparse inputs. This latent vector is then used as input to the decoder, which is responsible for reconstructing the complete body pose. More recently, methods based on the combination of transformers and generative algorithms have been introduced: In the same philosophy to *VAE-HMD*, full motion prior is used as a part of a pretraining process and a transformer-based encoder is then trained to predict the same motion latent as the full motion encoder using sparse inputs [41]. Additionally, *FLAG* [2] employs the technology of Normalizing Flow [38] that allows to compute *exact* pose likelihoods in contrast with VAEs and improved the quality of the full pose reconstruction in comparison with *VAE-HMD* outputs. *BoDiffusion* [9] employs a generative diffusion model [43] for motion synthesis to tackle this under-constrained reconstruction problem.

Finally, the full pose reconstruction from sparse inputs problem has been leveraged involving physical-simulation of the self-avatar’s body [27, 52]. These methods are built upon Reinforcement Learning for a physically plausible full-body pose reconstruction.

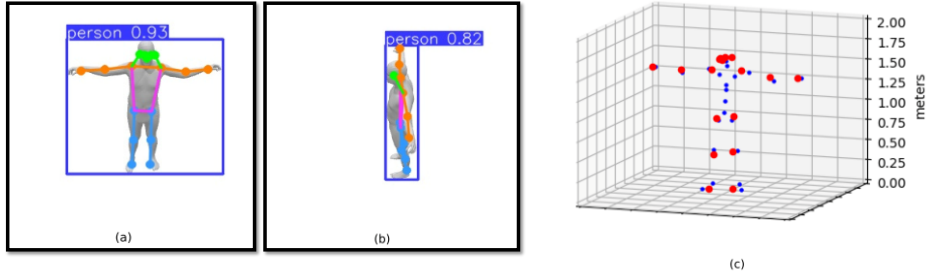


Figure 2: Reconstruction of the full-body position based on YOLOv8 algorithm. (a) and (b) Illustration of YOLOv8 detection applied to one frame of AMASS dataset rendered from two different perspectives. (c) the 3D Cartesian reconstruction pose computed by triangulation (red) compared with ground truth frame (blue)

2.2 Multimodal full-body pose estimation

In order to resolve the pose ambiguity, external motion sources have been deployed to extend the information tracked by the HMD and the handheld controllers.

Inspired by the success of 2D pose estimation from monocular RGB videos, *HybridTrack* [55] extends VR motion data with the 2D Cartesian coordinates. By combining an uncalibrated RGB camera for the lower body with inside-out tracking for the upper body, it offers cost-effective and user-friendly tracking capabilities.

Moreover, Liao et al. in [28] explore the fusion of sparse Inertial Measurement Units (IMUs) with a single RGB camera to achieve robust 3D human body reconstruction without the need for invasive multi-IMU setups or 2D joint detection. Using adaptive regression learning, a dual-stream network extracts features from IMUs and images, followed by a residual model-attention network that effectively fuses these features. The method significantly improves upon existing approaches by reducing errors in 3D joint positions and enhancing robustness, particularly in scenarios with occlusions or challenging environments.

Next, *EgoLocate* [56] emphasizes the importance of merging human motion sensing and environment sensing. They note that while human motion capture and environment sensing, such as SLAM, are crucial, they have traditionally been treated as separate domains. The authors introduce their *EgoLocate* system, which integrates sparse IMUs and a monocular camera to achieve real-time human motion capture, localization, and mapping.

Furthermore, Tome et al. in [48] propose a novel neural network architecture to address challenges such as strong perspective distortions, self-occlusion, a lack of labeled datasets, and the inherent ambiguity in lifting 2D joint positions to 3D space, demonstrating significant improvements in performance compared to existing methods, both in egocentric and front-facing camera scenarios.

Finally, *BodyTrak* [29] employs wrist-mounted cameras to capture informative images of body silhouettes and derives a working hypothesis that these partial body parts can effectively infer full-body poses. The contributions include the introduction of the first wrist-mounted device for full-body pose estimation, a custom deep learning pipeline, user studies, and discussions on the challenges and opportunities of applying such technology in practical contexts.

3 PROBLEM FORMULATION

The generation of the self-avatar’s full-body pose can be formalized as follow: considering a sequence of the sparse inputs $X_{0,\dots,T}$ and additional motion data $X_{0,\dots,T}^F$, a method Φ is designed to synthesize the full pose of the articulated self-avatar Y , as in Equation 1.

$$Y_T = \Phi(X_{0,\dots,T} \oplus X_{0,\dots,T}^F) \quad (1)$$

Similarly to *AvatarPoser* [20], $X_{0,\dots,T}$ gathers the Cartesian positions, the orientations, the linear and angular velocities of the head and the hands. We choose to extend the sparse inputs with the 3D Cartesian positions that can be estimated with a setup of one or multiple RGB(D) cameras [14]. This configuration has the advantage to not require any physical cumbersome tracking sensors on the user body which can infringe with its overall experience.

However, implementing this kind of solution is not a straightforward task since it deals with several constraints. In this context, the most common concerns are the lack of synchronization between the two motion sources, induced by a latency and a discrepancy between the data acquisition rate from both sources, joints occlusion and the low-fidelity of the Cartesian position estimation method.

- **Latency:** Depending on the cameras setup, a significant latency between the Cartesian positions and the sparse inputs can occur. Indeed, the head and hands positions and orientations are computed from inertial sensors or using SLAM-based algorithms [21] regarding the VR devices employed. The delay introduced by this process is relatively minor, typically ranging from 1 to 2 milliseconds when employing IMU-based head-mounted displays operating at 1000Hz for the rotational data and between 60Hz and 100Hz for the positional data. In contrast, the overall position estimation system often introduces a more substantial delay in the context of the deployed solution. Moreover, this latency is notably variable over time, primarily owing to frame loss. For instance, when assessing the *ZED 2* stereo camera [45] operating at a 60 FPS configuration, it exhibits a latency range of 30-45ms. This leads to desynchronization between the head-mounted display and the camera frames.
- **Framerate:** The motion data acquisition rates are not necessarily equivalents between the motion sources. The motion features of the HMD and the hands are extracted at 60Hz or 100Hz regarding the VR devices used [51]. While some cameras, such as the *ZED 2*, can also achieve frame rates of 60 or even 100 FPS, it’s crucial to take into account the computational demands associated with 3D position estimation [22]. This computational load can become substantial, contingent on the chosen algorithm and the number of individuals being tracked within the scene.
- **Occlusion:** It refers to the situation where a user being tracked is partially or completely hidden from the cameras used for tracking. This obstruction can occur when one user passes in front of another, or when a part of the user’s body blocks the view of one of its limbs. Occlusion poses a challenge for motion tracking systems because it can lose sight of one or multiple joints during the occluded period, leading to gaps

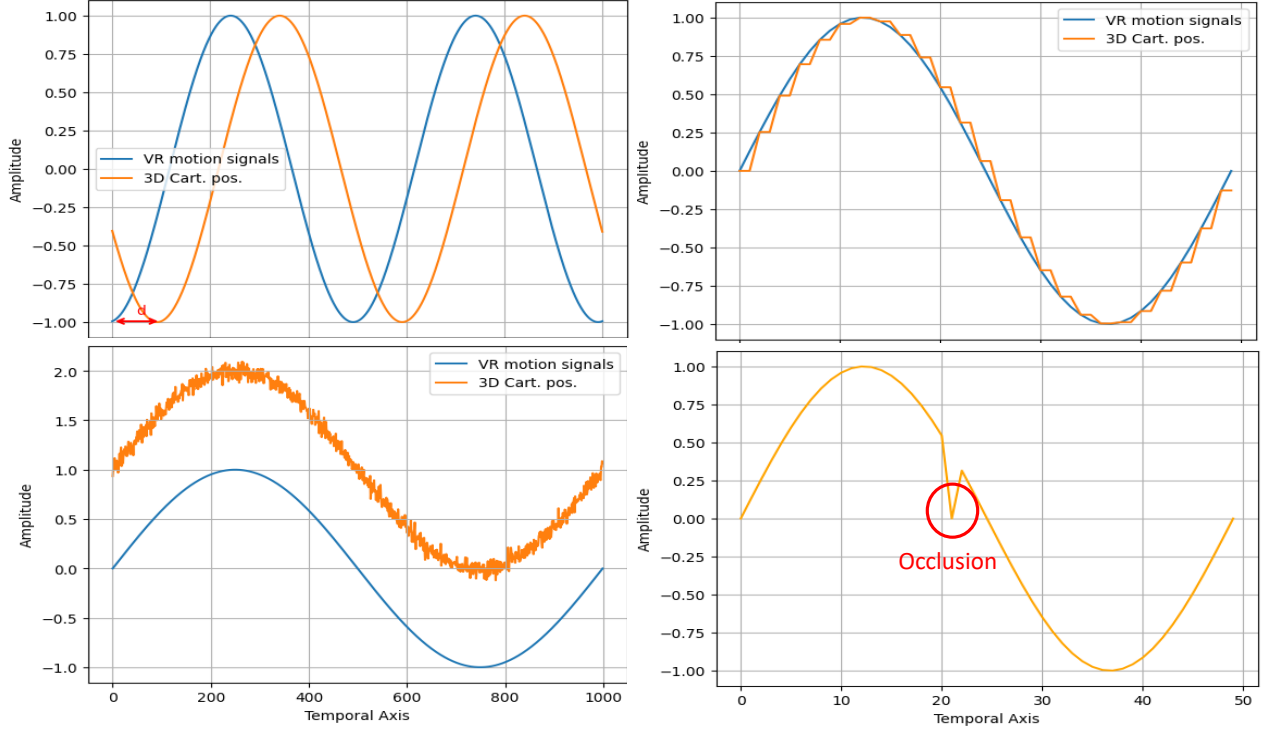


Figure 3: Examples of artifacts on motion signals. **Top left:** delay between two motion sources. **Top Right:** Cartesian position framerate reduced by $fps_{ratio} = 2$. **Bottom Left:** Gaussian noise applied on positional data. **Bottom Right:** Random occlusion *i.e.*, a joint position randomly set to zero.

or inaccuracies in the tracking pose. In our configuration, since we consider the additional motion information X^F as the 3D Cartesian positions estimated from cameras, our setup is subject to occlusion artifacts. This artifact gains prominence, particularly in scenarios involving multiple users interacting with each other, thereby increasing the likelihood of occluded joints.

- **Low accuracy:** Since the markerless motion tracking algorithm based on computer vision are often less accurate and reliable than optical markers, the low fidelity on the 3D Cartesian coordinates estimation can have a negative impact on the full-body pose reconstruction.

4 APPROACH

As in *AvatarPoser* [20], we rely on 3 subsets of the large-scale motion capture AMASS Dataset [31]: BMLrub [49], CMU [25] and HDM05 [32]. AMASS Dataset unifies optical-based motion capture datasets into a standard kinematic tree and use SMPL approach [30] to provide realistic 3D human meshes represented by a rigged body model. These subsets gather around 5200 motion samples for a duration of more than 20 hours. The standardized kinematic tree is structured into 22 joints.

In order to measure the impact of each artifacts individually, we propose to mimic the artifacts described in Section 3. Examples of these artifacts on motion signals are shown in Figure 3.

- We introduce a delay d between the full-body Cartesian positions and the sparse motion signals. In this configuration, Equation 1 becomes

$$Y_{T+d} = \Phi(X_{d,\dots,T+d} \oplus X_{0,\dots,T}^F) \quad (2)$$

- The framerate discrepancy between the two motion signals is tackled by quantifying $X_{0,\dots,T}^F$ using a framerate ratio fps_{ratio} between the two motion sources. The framerate of $X_{0,\dots,T}^F$, initially equivalent to $X_{0,\dots,T}$, is divided by fps_{ratio} .

- Inspired by the implementation of occlusion in [15], we define a probability σ_o of the joint being occluded for a given frame.

$$O_{0,\dots,T}^F \sim \text{Bernoulli}(\sigma_o) \quad (3)$$

$$X_{o_{0,\dots,T}}^F = X_{0,\dots,T}^F * O_{0,\dots,T}^F \quad (4)$$

$$Y_{o_T} = \Phi(X_{0,\dots,T} \oplus X_{o_{0,\dots,T}}^F) \quad (5)$$

- To mimic the low accuracy of the full-body position estimation, we add noise sampled from a zero-mean Gaussian distribution with a standard deviation σ . Increasing the standard deviation σ results in an augmented level of noise intensity within the Cartesian positions.

$$N_{0,\dots,T} \sim \mathcal{N}(0, \sigma^2) \quad (6)$$

$$X_{n_{0,\dots,T}}^F = X_{0,\dots,T}^F + N_{0,\dots,T} \quad (7)$$

$$Y_{n_T} = \Phi(X_{0,\dots,T} \oplus X_{n_{0,\dots,T}}^F) \quad (8)$$

However, this distribution may not accurately represent the actual noise in the markerless pose estimation system. To build a set of noisy poses that reflects the lack of accuracy of the data we could acquire in a real-case scenario, we use a solution based on the YOLOv8 algorithm [22]. As shown in Figure 2, using the SMPL+H model [40] linked to AMASS dataset, we define two virtual cameras to render the character as mesh from

two different viewpoints. The rendering resolution in pixels is 640X480. The processing of each rendered image with Yolo v8 detector (yolov8x-pose model) provides a pose as a set of characteristic points. We apply the triangulation function from the OpenCV library [5] to the characteristic points in the two perspectives, using the parameters of the virtual cameras as explained in [14]. The result is a 3-dimensional reconstruction of the pose computed for each frame in the dataset.

5 EXPERIMENTS

For our experiments, we train the animation model Φ with $X_{0,\dots,T} \oplus X_{0,\dots,T}^F$ as input where $X_{0,\dots,T}^F$ is a sequence of the ground truth 3D Cartesian positions and then measure the impact of each artifacts independently. We rely on *AvatarPoser* [20] and an adapted version of *HybridTrack* [55], two state-of-the-art deep learning-based models for the animation of the self-avatar’s full-body. *AvatarPoser* [20] is a model built upon the Transformer architecture that encodes the sparse inputs into a high-level complex motion representation to estimate the full-body local orientations as well as the global displacement of the root. Providing 3D Cartesian positions into this model helps to resolve the pose ambiguity that can arise due to the lack of lower body motion information. *HybridTrack* [55] employs a CNN-1D based architecture inspired by the method in [35], that is fed by the sparse information and a single-view of 2D Cartesian coordinates to generate the full body pose. In our experiments, we adapt this model to our requirements by providing the 3D Cartesian positions instead.

We add a latency between the Cartesian positions and the sparse inputs of $d = 2, 4$ or 6 frames following the delays discussed in Section 3. Then, since the framerate is sensitive to the computational load induced by several factors such as the number of cameras or the number of tracked users, we divided the Cartesian coordinates framerate by a factor $fps_{ratio} = 2, 3$ or 4. In this last case, if the motion data acquisition rate from VR devices reaches 100Hz, the cameras achieve a framerate of 25Hz. Regarding the spatial-temporal noise, we set up $\sigma_o = 0.01$ and 0.05 of occlusion probability ($\sigma_o = 0.1$ in [15]) and $\sigma = 1cm, 2cm$ and $5cm$ which are reasonable noise levels considering the error reconstruction in multi-users 3D pose estimation tracking from monocular videos [58] (60mm on *Human36M* dataset [17]).

Finally, we test our implementations in a real-case scenario: from AMASS Dataset, we use the pose estimation from *YOLOv8* to gather the 3D Cartesian positions, instead of using the ground truth Cartesian positions to train and to evaluate a specific animation model.

We consider a temporal window of 40 previous frames and the current frame ($T = 41$ frames) feeding the models. Both of them are trained during 10k epochs on a NVIDIA GTX1080 GPU, following the training procedure and hyperparameters in *AvatarPoser* [20]¹.

6 RESULTS

Table 1 shows the results of the evaluation of the two animation models to the considered motion signals artifacts. We rely on 3 metrics called the Mean Per Joint Positional/Rotational/Velocity Error, respectively referred as *MPJPE*, *MPJRE* and *MPJVE*. It indicates the deviation between the avatar’s full pose estimated by the model and the ground truth pose. We also refer in the aforementioned Table the results on the full pose reconstruction from $X_{0,\dots,T}$ for *AvatarPoser* but not for *HybridTrack* since it has not been designed to tackle the problematic of the full pose reconstruction from only the sparse inputs.

We observe that the overall behavior is that the errors increase with the intensity of the artifact for both models and regardless the

evaluated artifact. The models trained with the clean and synchronized data from AMASS dataset does not encompass the issues related to the animation of the self-avatar’s character based on the sparse inputs and the Cartesian positions estimated from cameras. Then, when the spatio-temporal motion artifacts, *i.e.*, the occlusions and the noise, increases in intensity, *HybridTrack* provides more accurate full pose reconstruction than *AvatarPoser*. In this configuration, *HybridTrack* exhibits greater resilience and accuracy than *AvatarPoser*, showcasing its enhanced performance in challenging conditions marked by heightened motion artifacts. However, it is deemed ineffective to incorporate 3D Cartesian positions in both animation model when confronted with intense artifacts. Indeed, while *HybridTrack* effectively mitigates the impact of these artifacts, *AvatarPoser*, trained solely with $X_{0,\dots,T}$, outperforms *HybridTrack* across various configurations. We highlighted in Table 1 the configurations where the models fed by $X_{0,\dots,T} \oplus X_{0,\dots,T}^F$ exhibits more accurate pose reconstruction than *AvatarPoser* trained with sparse inputs. The tested artifacts appear to have a more pronounced effect on velocity compared to positions and orientations. In summary, this evaluation highlights the sensitivity of the models to variations in the training parameters, limiting their deployment in less controlled environments.

The results, reported in 4, show the comparison between models trained with ground truth and models trained with Cartesian positions generated with YOLO. *HybridTrack* produces less accurate motion reconstruction than *AvatarPoser* with solely the sparse inputs regarding the upper body. However, *AvatarPoser* trained with the 3D coordinates estimated from YOLOv8 outperforms *AvatarPoser* from sparse inputs, except for the velocity of the upper body region. Comparing these two cases, we can also observe that we do not get any significant difference in velocity in the lower body region either. So, there is no clear improvement in terms of noise present in the synthesized movement.

Illustrative pose samples are depicted in Figure 5. The coloration on various regions of the mesh represents the positional error specific to each region. The figure serves to highlight that the quality of motion reconstruction diminishes as noise is introduced to the 3D coordinates. More videos presenting motion samples can be found here²

7 LIMITATIONS AND PERSPECTIVES

First of all, our analyses emphasize the sensitivity of *AvatarPoser* and *HybridTrack* when trained with ground truth Cartesian positions and sparse inputs from VR devices. While it has been shown that incorporating Cartesian positions aids both models in addressing this challenge, the artifacts examined in this study—such as latency between motion sources, discrepancies in framerate, low accuracy of pose reconstruction, and occlusions—result in a degradation of the quality of the self-avatar’s full-body pose reconstruction.

Then, even if we consider these effects in the training process, both of the models failed to improve the upper-body velocities in comparison to the model trained only with the sparse inputs. Mitigating this effect will be crucial for further development, especially when several users share the scene captured by the cameras. Indeed, in this context, the risk of major artifacts such as occlusions can explode.

Finally, considering the desynchronization between the two motion signals, recent methods have been proposed to integrate temporally sparse observations in Transformer within a medical context [47]. We believe that the integration of such systems will benefit the field of self-avatar’s animation regarding the issues discussed in this work.

¹<https://github.com/eth-siplab/AvatarPoser.git>

²<https://figshare.com/s/a6f9c9770e4be4919230>

Table 1: Evaluation of the sensitivity to common artifacts in the context of the self-avatar’s animation from sparse inputs and 3D Cartesian coordinates. The highlighted results refer to configurations that outperform *AvatarPoser* with sparse inputs. We observe that *AvatarPoser* overall gives the best results when it comes to deal with clean motion data. However, as a matter of example, when the noise or occlusion intensity increases, *HybridTrack* leads to lower reconstruction error.

	Body subset	<i>AvatarPoser</i>			<i>HybridTrack</i>		
		MPJPE (cm) ↓	MPJRE (°) ↓	MPJVE (cm/s) ↓	MPJPE (cm) ↓	MPJRE (°) ↓	MPJVE (cm/s) ↓
Sparse inputs	Up	1.65	5.64	12.86	-	-	-
Sparse inputs	Low	6.79	6.4	44.35	-	-	-
GT 3D Cart. Pos.	Up	0.72	2.52	8.1	2.38	6.51	20.72
GT 3D Cart. Pos.	Low	1.41	2.03	14.19	4.97	5.55	40.81
Noise $\sigma(cm)$							
1	Up	1.09	3.89	60.78	2.39	6.58	35.39
1	Low	1.96	3.37	107.91	5.02	5.61	69.86
2	Up	1.74	6.33	121.98	2.46	6.78	55.94
2	Low	3.04	5.65	215.17	5.18	5.77	111.11
5	Up	4.18	15.04	326.92	2.92	8.13	118.89
5	Low	7.2	13.35	565.61	6.25	6.83	240.02
Occlusion σ_o							
0.01	Up	3.05	8.18	272.48	4.05	10.16	231.65
0.01	Low	5.87	5.26	525.83	7.87	7.59	414.22
0.05	Up	10.36	23.47	912.65	8.95	19.14	525.12
0.05	Low	19.68	13.73	1756.09	15.82	13.13	992.5
Framerate ratio fps_{ratio}							
2	Up	0.87	2.73	23.98	2.43	6.54	21.96
2	Low	1.81	2.41	57.02	5.19	5.71	45.29
3	Up	1.05	3.03	30.91	2.49	6.6	27.71
3	Low	2.22	2.81	74.18	5.44	5.89	68.37
4	Up	1.22	3.35	34.47	2.55	6.66	25.88
4	Low	2.64	3.21	82.65	5.72	6.1	57.54
Delay d (frames)							
2	Up	1.44	3.78	14.36	2.62	6.73	21.77
2	Low	3.02	7.23	24.57	5.98	6.31	47.5
4	Up	2.24	5.05	19.8	2.95	7.12	23.28
4	Low	4.83	5.42	42.69	7.26	7.31	56.27
6	Up	3.02	7.23	24.57	3.34	7.63	25.22
6	Low	6.5	7.01	54.15	8.66	8.36	65.51

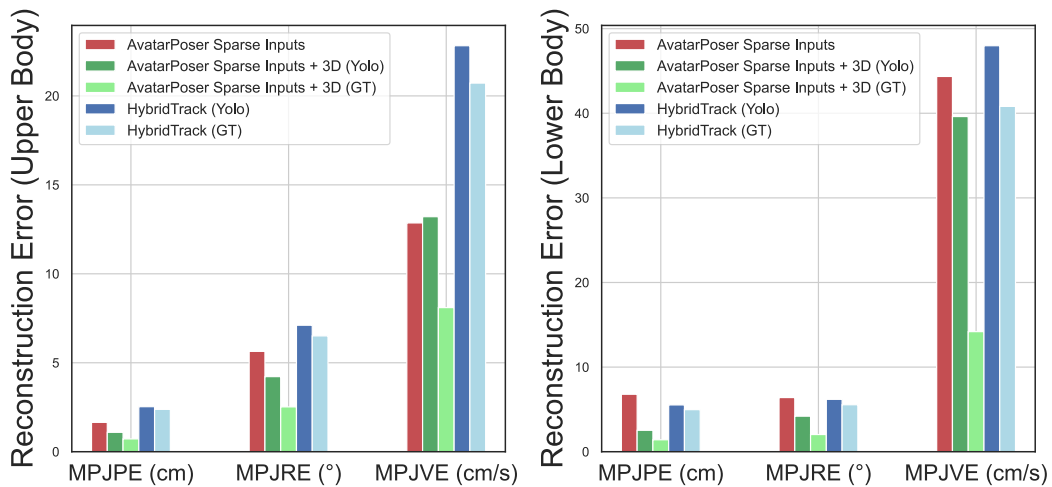


Figure 4: Reconstruction errors regarding the models trained with ground truth and Cartesian positions from YOLOv8.

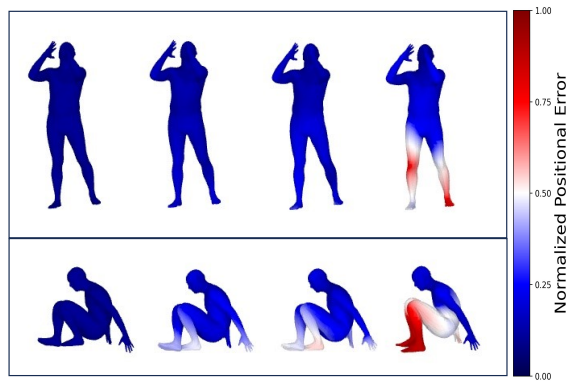


Figure 5: Illustration of two pose samples derived from the ground truth data (**Left**). In the **Mid-Left** image, *AvatarPoser* is trained with ground truth 3D Cartesian positions and provided with a sequence of this ground truth. The **Mid-Right** image displays the effects of introducing Gaussian noise into the 3D Cartesian coordinates, with noise levels $\sigma = 0.01$. **Right**: *AvatarPoser* trained with only sparse inputs. The positional features are improved in comparison to those produced by solely the sparse inputs, even when the Cartesian coordinates are degraded with a Gaussian noise with $\sigma = 1\text{cm}$.

8 CONCLUSION

In this work, we conducted experiments in the context of the self-avatar's full-body pose reconstruction from the head and hands motion features and the 3D Cartesian coordinates. This additional motion data has been chosen so that it can be extracted from a non-intrusive systems, such as RGB-D cameras, to guarantee the user experience. First, we discussed about the major concerns that need to be taken into account in the design of such system. The desynchronization between the motion signals from the VR devices and the Cartesian positions, as well as the spatio-temporal motion artifacts such as noise and occlusions are artifacts that importantly degrades the reliability of the animation system. We showed that the precision of the pose estimation system is crucial in this context, especially considering the velocity of the reconstructed upper body. From the results of these analyzes, we believe that this work provides valuable insights concerning the task of self-avatar's animation based on multimodal data.

ACKNOWLEDGMENTS

This research is supported and funded by TRAIL Institute. S.A. Ghasemzadeh is funded by FRIA/FNRS.

REFERENCES

- [1] K. Ahuja, E. Ofek, M. Gonzalez-Franco, C. Holz, and A. D. Wilson. Coolmoves: User motion accentuation in virtual reality. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 5(2):1–23, 2021.
- [2] S. Aliakbarian, P. Cameron, F. Bogo, A. Fitzgibbon, and T. J. Cashman. Flag: Flow-based 3d avatar generation from sparse observations, 2022.
- [3] A. K. Bashabsheh, H. H. Alzoubi, and M. Z. Ali. The application of virtual reality technology in architectural pedagogy for building constructions. *Alexandria Engineering Journal*, 58(2):713–723, 2019.
- [4] M.-S. Bracq, E. Michinov, and P. Jannin. Virtual reality simulation in nontechnical skills training for healthcare professionals: a systematic review. *Simulation in Healthcare*, 14(3):188–194, 2019.
- [5] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.
- [6] M. Büttner and S. Clavet. Motion matching—the road to next gen animation. *Proc. of Nucl. ai*, 1(2015):2, 2015.

- [7] Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh. Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.
- [8] P. Caserman, P. Achenbach, and S. Gobel. Analysis of inverse kinematics solutions for full-body reconstruction in virtual reality. pp. 1–8, 08 2019. doi: 10.1109/SeGAH.2019.8882429
- [9] A. Castillo, M. Escobar, G. Jeanneret, A. Pumarola, P. Arbeláez, A. Thabet, and A. Sanakoyeu. Bodiffusion: Diffusing sparse observations for full-body human motion synthesis. *arXiv preprint arXiv:2304.11118*, 2023.
- [10] D. Checa and A. Bustillo. A review of immersive virtual reality serious games to enhance learning and training. *Multimedia Tools and Applications*, 79:5501–5527, 2020.
- [11] A. Dittadi, S. Dziadzio, D. Cosker, B. Lundell, T. J. Cashman, and J. Shotton. 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.
- [12] T. Dorta, G. Kinayoglu, and M. Hoffmann. Hyve-3d and the 3d cursor: Architectural co-design with freedom in virtual reality. *International Journal of Architectural Computing*, 14(2):87–102, 2016.
- [13] J. H. Geissinger and A. T. Asbeck. Motion inference using sparse inertial sensors, self-supervised learning, and a new dataset of unscripted human motion. *Sensors*, 20(21), 2020. doi: 10.3390/s20216330
- [14] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, New York, NY, USA, 2 ed., 2003.
- [15] D. Holden. Robust solving of optical motion capture data by denoising. *ACM Transactions on Graphics (TOG)*, 37(4):1–12, 2018.
- [16] Y. Huang, M. Kaufmann, E. Aksan, M. J. Black, O. Hilliges, and G. Pons-Moll. Deep inertial poser: Learning to reconstruct human pose from sparse inertial measurements in real time. *ACM Transactions on Graphics (TOG)*, 37(6):1–15, 2018.
- [17] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1325–1339, jul 2014.
- [18] L. Jensen and F. Konradsen. A review of the use of virtual reality head-mounted displays in education and training. *Education and Information Technologies*, 23:1515–1529, 2018.
- [19] F. Jiang, X. Yang, and L. Feng. Real-time full-body motion reconstruction and recognition for off-the-shelf vr devices. In *Proceedings of the 15th ACM SIGGRAPH Conference on Virtual-Reality Continuum and Its Applications in Industry-Volume 1*, pp. 309–318, 2016.
- [20] J. Jiang, P. Strelci, H. Qiu, A. Fender, L. Laich, P. Snape, and C. Holz. Avatarposer: Articulated full-body pose tracking from sparse motion sensing. In *Proceedings of European Conference on Computer Vision*. Springer, 2022.
- [21] L. Jinyu, Y. Bangbang, C. Danpeng, W. Nan, Z. Guofeng, and B. Hujun. Survey and evaluation of monocular visual-inertial slam algorithms for augmented reality. *Virtual Reality & Intelligent Hardware*, 1(4):386–410, 2019.
- [22] G. Jocher, A. Chaurasia, and J. Qiu. YOLO by Ultralytics, Jan. 2023.
- [23] J. Kim, Y. Seol, and J. Lee. Realtime performance animation using sparse 3d motion sensors. In *Motion in Games: 5th International Conference, MIG 2012, Rennes, France, November 15-17, 2012. Proceedings 5*, pp. 31–42. Springer, 2012.
- [24] M. Kim and S. Lee. Fusion poser: 3d human pose estimation using sparse imus and head trackers in real time. *Sensors*, 22(13), 2022. doi: 10.3390/s22134846
- [25] C. G. Lab. Cmu graphics lab motion capture database., 2000.
- [26] P. Lang. Inverse kinematics in dead and buried, 2016.
- [27] S. Lee, S. Starke, Y. Ye, J. Won, and A. Winkler. Questensvim: Environment-aware simulated motion tracking from sparse sensors. *arXiv preprint arXiv:2306.05666*, 2023.
- [28] X. Liao, J. Zhuang, Z. Liu, J. Dong, K. Song, and J. Xiao. Reconstructing 3d human pose and shape from a single image and sparse imus. *PeerJ Comput. Sci.*, 2023.
- [29] H. Lim, Y. Li, M. Dressa, F. Hu, J. H. Kim, R. Zhang, and C. Zhang. Bodytrak: Inferring full-body poses from body silhouettes using a

- miniature camera on a wristband. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 6(3), sep 2022. doi: 10.1145/3552312
- [30] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black. Smpl: A skinned multi-person linear model. *ACM transactions on graphics (TOG)*, 34(6):1–16, 2015.
- [31] N. Mahmood, N. Ghorbani, N. F. Troje, G. Pons-Moll, and M. J. Black. AMASS: Archive of motion capture as surface shapes. In *International Conference on Computer Vision*, pp. 5442–5451, Oct. 2019.
- [32] M. Müller, T. Röder, M. Clausen, B. Eberhardt, B. Krüger, and A. Weber. Documentation mocap database hdm05. *Computer Graphics Technical Report CG-2007-2, Universität Bonn*, 2007.
- [33] Y. Pan and A. Steed. The impact of self-avatars on trust and collaboration in shared virtual environments. *PLOS ONE*, 12(12):1–20, 12 2017. doi: 10.1371/journal.pone.0189078
- [34] Y. Pan and A. Steed. How foot tracking matters: The impact of an animated self-avatar on interaction, embodiment and presence in shared virtual environments. *Frontiers in Robotics and AI*, 6, 2019. doi: 10.3389/frobt.2019.00104
- [35] D. Pavlo, C. Feichtenhofer, D. Grangier, and M. Auli. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 7753–7762, 2019.
- [36] A. S. Pillai and P. S. Mathew. Impact of virtual reality in healthcare: a review. *Virtual and augmented reality in mental health treatment*, pp. 17–31, 2019.
- [37] J. L. Ponton, H. Yun, C. Andujar, and N. Pelechano. Combining motion matching and orientation prediction to animate avatars for consumer-grade vr devices. In *Computer Graphics Forum*, vol. 41, pp. 107–118. Wiley Online Library, 2022.
- [38] D. Rezende and S. Mohamed. Variational inference with normalizing flows. In *International conference on machine learning*, pp. 1530–1538. PMLR, 2015.
- [39] S. L. Rogers, R. Broadbent, J. Brown, A. Fraser, and C. P. Speelman. Realistic motion avatars are the future for social interaction in virtual reality. *Frontiers in Virtual Reality*, 2, 2022. doi: 10.3389/frvir.2021.750729
- [40] J. Romero, D. Tzionas, and M. J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 36(6), Nov. 2017.
- [41] M. Shin, D. Lee, and I.-K. Lee. Utilizing task-generic motion prior to recover full-body motion from very sparse signals. *arXiv preprint arXiv:2308.15839*, 2023.
- [42] P. Smutny, M. Babiuch, and P. Foltyněk. A review of the virtual reality applications in education and training. In *2019 20th International Carpathian Control Conference (ICCC)*, pp. 1–4. IEEE, 2019.
- [43] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pp. 2256–2265. PMLR, 2015.
- [44] A. Steed, Y. Pan, F. Zisch, and W. Steptoe. The impact of a self-avatar on cognitive load in immersive virtual reality. In *2016 IEEE Virtual Reality (VR)*, pp. 67–76, 2016. doi: 10.1109/VR.2016.7504689
- [45] Stereolabs. Zed 2. <https://www.stereolabs.com/zed-2/>.
- [46] Z. Tan, Y. Hu, and K. Xu. Virtual reality based immersive telepresence system for remote conversation and collaboration. In *Next Generation Computer Animation Techniques: Third International Workshop, AniNex 2017, Bournemouth, UK, June 22-23, 2017, Revised Selected Papers 3*, pp. 234–247. Springer, 2017.
- [47] S. Tipirmeni and C. K. Reddy. Self-supervised transformer for sparse and irregularly sampled multivariate clinical time-series. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 16(6):1–17, 2022.
- [48] D. Tome, T. Alldieck, P. Peluse, G. Pons-Moll, L. Agapito, H. Badino, and F. de la Torre. Selfpose: 3d egocentric pose estimation from a headset mounted camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(06):6794–6806, jun 2023. doi: 10.1109/TPAMI.2020.3029700
- [49] N. F. Troje. Decomposing biological motion: A framework for analysis and synthesis of human gait patterns. *Journal of vision*, 2(5):2–2, 2002.
- [50] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.
- [51] VR-Compare. Vr hardware comparison. <https://vr-compare.com/compare?h1=pDTZ02PkT&h2=mLbW9G7f4&h3=0jLuwg808-j>.
- [52] A. Winkler, J. Won, and Y. Ye. Questsim: Human motion tracking from sparse sensors with simulated avatars. In *SIGGRAPH Asia 2022 Conference Papers*, pp. 1–8, 2022.
- [53] Y. Wu, Y. Wang, S. Jung, S. Hoermann, and R. W. Lindeman. Towards an articulated avatar in vr: Improving body and hand tracking using only depth cameras. *Entertainment Computing*, 31:100303, 2019. doi: 10.1016/j.entcom.2019.100303
- [54] D. Yang, D. Kim, and S.-H. Lee. 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.
- [55] J. Yang, T. Chen, F. Qin, M. S. Lam, and J. A. Landay. Hybridtrak: adding full-body tracking to vr using an off-the-shelf webcam. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2022.
- [56] X. Yi, Y. Zhou, M. Habermann, V. Golyanik, S. Pan, C. Theobalt, and F. Xu. EgoLocate: Real-time motion capture, localization, and mapping with sparse body-mounted sensors. *arXiv preprint arXiv:2305.01599*, 2023.
- [57] X. Yi, Y. Zhou, M. Habermann, S. Shimada, V. Golyanik, C. Theobalt, and F. Xu. 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 (CVPR)*, pp. 13167–13178, June 2022.
- [58] A. Zanfir, E. Marinoiu, M. Zanfir, A.-I. Popa, and C. Sminchisescu. Deep network for the integrated 3d sensing of multiple people in natural images. *Advances in neural information processing systems*, 31, 2018.
- [59] X. Zhang, X. Chen, X. Dai, and X. Di. Dual attention poser: Dual path body tracking based on attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2794–2803, 2023.