Pseudo-label based unsupervised fine-tuning of a monocular 3D pose estimation model for sports motions

Tomohiro Suzuki Ryota Tanaka Kazuya Takeda Keisuke Fujii
Nagoya University, Japan

Abstract

Accurate motion capture is useful for sports motion analysis, but requires higher acquisition costs. Monocular or few camera multi-view pose estimation provides an accessible but less accurate alternative, especially for sports motion, due to training on datasets of daily activities. In addition, multi-view estimation is still costly due to camera calibration. Therefore, it is desirable to develop an accurate and cost-effective motion capture system for the daily training in sports. In this paper, we propose an accurate and convenient sports motion capture system based on unsupervised fine-tuning. The proposed system estimates 3D joint positions by multi-view estimation based on automatic calibration with the human body. These results are used as pseudo-labels for fine-tuning of the recent higher performance monocular 3D pose estimation model. Since the fine-tuning improves the model accuracy for sports motion, we can choose multi-view or monocular estimation depending on the situation. We evaluated the system using a running motion dataset and ASPset-510, and showed that fine-tuning improved the performance of monocular estimation to the same level as that of multi-view estimation for running motion. Our proposed system can be useful for the daily motion analysis in sports.

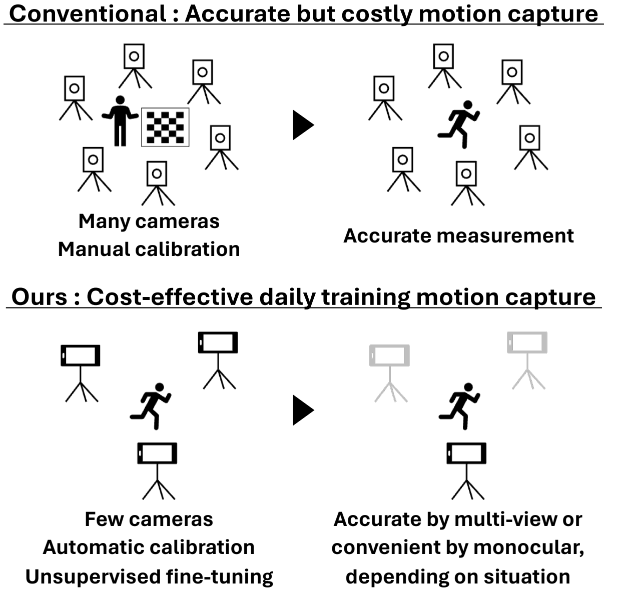


Figure 1. An overview of our proposed system. Conventional motion capture requires many specialized cameras and calibration costs. Our system can directly estimate 3D keypoints with few (smartphone) cameras based on automatic calibration, and improve the estimation performance through unsupervised fine-tuning. After fine-tuning, we can choose monocular or multi-view estimation depending on the training situation.

1. Introduction

In many sports, it is important for athletes to capture motion data to obtain useful information and objectively evaluate movement. To accurately capture motion data, conventional markerless motion capture techniques use a large number of cameras to estimate 3D joint positions. However, it is difficult to prepare many cameras for daily use. In addition, a camera calibration process that determines the camera's position in the world coordinate system is time-consuming. To address the above problems, some monocular 2D or 3D pose estimation [1–3, 13, 15, 19, 22, 23, 31, 33] including mesh estimation [5, 11], multi-view pose estimation with few cameras [6, 21, 29], and pose estimation using IMUs [32] are proposed. Since these approaches allow low-cost measurement, they are applied to some sports to detect faults in race walking [24, 25] and edge errors in figure skating [27, 28], and motion analysis for martial arts [4, 18] and gymnastics [10].

Monocular pose estimation is a highly developed technique in computer vision for estimating 2D / 3D joint positions (keypoints) from a single-view image or video. The 2D pose estimation models [1, 2, 23, 31] estimate two-dimensional keypoints in the image pixel coordinate sys-

Email: {suzuki.tomohiro, ryota.tanaka}@g.sp.m.is.nagoya-u.ac.jp, kazuya.takeda@nagoya-u.jp, fujii@i.nagoya-u.ac.jp

tem, and the 3D pose estimation models [3, 13, 19, 22, 33] estimates three-dimensional keypoints in the camera coordinate system from the sequence of the 2D keypoints. These are useful to get keypoints and calculate motion data easily. However, most available pose estimation datasets [8, 9, 14, 30] are for daily activity motion such as walking, standing, and cooking, which makes it difficult to apply the pre-trained pose estimation models to sports. Therefore, pre-trained models using such datasets need to be fine-tuned using annotated sports motion data for application [24, 25], but the annotation cost is expensive for athletes. Moreover, current monocular 3D pose estimation cannot accurately estimate the scale of the target, and it is also difficult to estimate the translation of the motion. For example, in [3, 19, 22, 33], the scale of the person in the dataset is ignored in the data pre-processing, and the models estimate the depth relative to the root joint.

A different approach from the monocular pose estimation is the attempt to estimate 3D keypoints and motion data from multi-view videos [6, 21, 29]. Since multiple cameras can solve the occlusion problem, the accuracy of multi-view estimation is higher than that of monocular estimation. This approach can also estimate the absolute depth of the person, which is useful for sports motion analysis. On the other hand, an accurate multi-view system such as Opencap [29] requires a camera calibration similar to markerless motion capture. Such a calibration process is desirable for automation to improve the convenience of daily motion analysis, as it takes extra time for measurements. To avoid calibration, Ingwersen et al. [6] train the monocular 3D pose estimation model using multi-view consistency loss, which has constraints on camera positioning to take advantage of consistency loss. As a result, the applicability and convenience of measurements may be limited.

For automatic calibration, some studies [12, 16, 20, 26] use the moving human body as a calibration target. In particular, Lee et al. [12] combine the results of multi-view monocular 3D pose estimation to automatically compute extrinsic parameters of the cameras. After calibration, they attempt unsupervised fine-tuning of the monocular 3D pose estimation model using more accurate keypoint coordinates obtained from the calibration result. This study evaluates daily activities such as walking, and it is not obvious that their method is effective for sports motion. If this technique could be applied to sports motion, it would be useful to create pose estimation models optimized for sports motion, which is important for motion analysis.

In this paper, we propose an accurate and convenient motion capture (especially pose estimation) system for sports. Our research aims to realize cost-effective 3D pose estimation that can be easily measured with simple equipment for daily sports training. An overview of our research is shown in Figure 1. Compared to conventional motion capture, which requires many cameras, our proposed system does not require many cameras and an explicit camera calibration process, and is easier to measure. In addition, since it can optimize the monocular 3D pose estimation model for sports motion with unsupervised fine-tuning, we can choose multi-view or monocular estimation depending on the training situation. In the system, we first estimate 3D keypoints using multi-view monocular pose estimation results from automatically calibrated camera videos. For the automatic calibration, we use the previous approach [12]. The coordinates of the keypoints obtained from the multi-view 3D pose estimation are then used as pseudo-labels to fine-tune the monocular 3D pose estimation model. Many pre-trained monocular 3D pose estimation models are not optimized for sports motion, but in this study it is optimized for sports by unsupervised fine-tuning without annotation cost. Multi-view pose estimation is required for fine-tuning, but does not require a large number of cameras, and there are no strict constraints on placement. After unsupervised fine-tuning, we can also use the fine-tuned monocular 3D pose estimation model alone. This costless and accurate pose estimation system has great advantages for daily training use, especially for amateurs. We evaluate the system on the original running motion dataset and the Australian Sports Pose Dataset (ASPset-510) [17], which contain more intense motion than typical motion datasets, and show that the proposed method is useful for convenience sports motion analysis. Our running motion dataset consists of videos captured with a smartphone camera, showing that the proposed system can also be used with simple video cameras.

To summarize, our contributions are:

1. We propose a cost-effective single or multi-view motion capture system optimized for sports motion with unsupervised fine-tuning. It utilizes automatic calibration with the human body and can support daily sports training analysis.
2. We demonstrate that our proposed system can improve the accuracy of 3D pose estimation for sports motion with no annotation cost.
3. We show that the monocular 3D pose estimation after fine-tuning is comparable to multi-view, especially for a running motion dataset.

2. Methods

Our goal is to realize a cost-effective motion capture system optimized for sports motion with unsupervised fine-tuning of the monocular 3D pose estimation model using pseudo-labels. Pseudo-labels are 3D keypoint coordinates obtained from automatically calibrated multi-view videos using monocular 2D and 3D pose estimation models. Figure 2 shows the process of training the monocular 3D pose estimation model using the unsupervised method. The process can be divided into three steps. First, multi-view videos

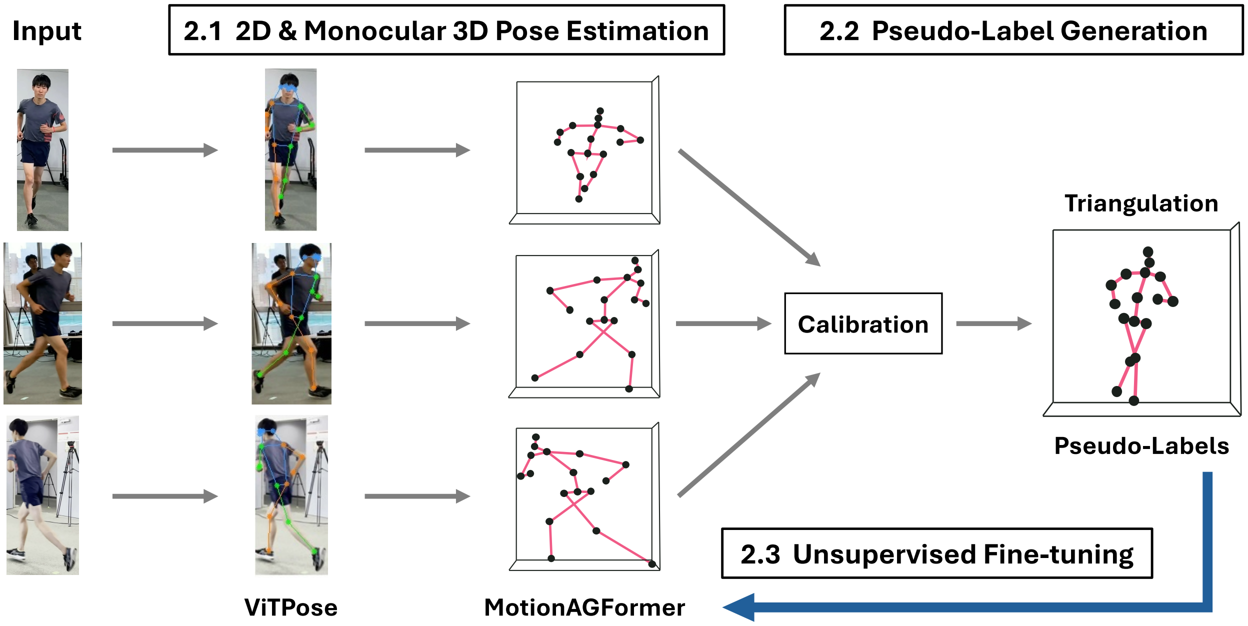

Input
2.1 2D & Monocular 3D Pose Estimation
2.2 Pseudo-Label Generation
Triangulation
Calibration
Pseudo-Labels
2.3 Unsupervised Fine-tuning
ViTPose
MotionAGFormer

Figure 2. Our proposed system process. In Section 2.1, we describe monocular pose estimation for multi-view pose estimation based on automatic calibration. In Section 2.2, we describe pseudo-label generation using automatic calibration results. In Section 2.3, we describe the unsupervised fine-tuning of the monocular 3D pose estimation model to optimize it for target sports motion.

are separately input into the 2D pose estimation model, and 2D keypoints of persons in each video are estimated. We use ViTPose [31] as the 2D pose estimation model as described in Section 2.1. The 2D keypoints are input into the monocular 3D pose estimation model, and 3D keypoints of the person are estimated. We use MotionAGFormer [22] as the 3D pose estimation model.

Next, pseudo-labels are generated by triangulating the 2D keypoints with extrinsic camera parameters obtained from automatic camera calibration, which is described in Section 2.2. The extrinsic camera parameters are calibrated using 3D keypoints from each video. In this step, we automatically calibrate the extrinsic parameters (camera rotation and translation) using the previous study method [12]. After calibration, more accurate 3D keypoints are estimated by triangulation using extrinsic parameters and 2D keypoints from each camera view. Finally, the monocular 3D pose estimation model is fine-tuned using triangulated 3D keypoints as pseudo-labels as described in Section 2.3. The monocular 3D pose estimation model pre-trained by general pose datasets cannot perform well for sports motion. Since we do not use labeled data in the whole process, our proposed method is effective to optimize the model for sports motion with low cost. After fine-tuning, we can use the monocular 3D pose estimation model alone or the more accurate multi-view estimation, depending on the situation. In our research, we use the original running motion dataset and Australian Sports Pose Dataset (ASPset-510) [17] for the experiments. We did not use other sports motion datasets, such as SportsPose dataset [7] because it does not provide multi-view data.

2.1. 2D & Monocular 3D Pose Estimation

To estimate 2D keypoints from videos, we use ViTPose [31], which is pre-trained with the COCO keypoint dataset [14]. ViTPose is the state-of-the-art model for this dataset. Since ViTPose is the top-down pose estimation model, we detect the bounding box of the persons and estimate 2D keypoints of each person.

For monocular 3D pose estimation, we use MotionAGFormer [22], which is pre-trained with the Human3.6M dataset [8]. In monocular 3D pose estimation, 2D keypoints are input into the model and 3D keypoints are estimated by exploiting spatial and temporal features. MotionAGFormer achieves fast and accurate 3D pose estimation by capturing global features of 2D keypoints with the Transformer and local features with a graph convolutional network. However, in ASPset-510, the size of the person in the image is much smaller than that in the Human3.6M dataset, and the pre-trained model does not estimate well. Therefore, for ASPset-510, we augment the Human3.6M dataset by randomly scaling the input 2D keypoints to a smaller size, creating a pre-trained model that is different from the one pre-trained in the MotionAGFormer paper (see Section 3.2 for more details).

2.2. Pseudo-Label Generation

When generating pseudo-labels, we first automatically calibrate the cameras using the previous study method [12] to obtain extrinsic camera parameters. Before automatic calibration, we manually calibrate intrinsic camera parameters $\boldsymbol{K}$. Since the intrinsic parameters are camera-specific, we only need to calibrate them once.

The purpose of this calibration method is to estimate the camera rotation $\boldsymbol{R}$ and translation $\boldsymbol{t}$ using monocular 3D pose estimation results. Following the previous method [12], 3D keypoints from each monocular 3D pose estimation are considered as "oriented points", which are the set of point coordinates $\boldsymbol{x} \in \mathbb{R}^3$ and orientation $\boldsymbol{v} \in \mathbb{R}^3$. The oriented point $\langle \boldsymbol{x}_i^c, \boldsymbol{v}_i^c \rangle$ in the coordinate system of camera c is

$$\begin{aligned} \boldsymbol{x}_i^c &= \boldsymbol{R}^c \boldsymbol{x}_i + \boldsymbol{t}^c, && (1) \\ \boldsymbol{v}_i^c &= \boldsymbol{R}^c \boldsymbol{v}_i, && (2) \end{aligned}$$

where i ($i = 1, \ldots, N$) is the keypoint index, $\boldsymbol{R}^c$ is the rotation of camera c, and $\boldsymbol{t}^c$ is the translation of camera c. The point $\boldsymbol{y}_i^c \in \mathbb{R}^2$ projected on the image of camera c is

$$\lambda_i^c \begin{bmatrix} \boldsymbol{y}_i^c \\ 1 \end{bmatrix} = \boldsymbol{K}^c \boldsymbol{x}_i^c = \boldsymbol{K}^c \left(\boldsymbol{R}^c \boldsymbol{x}_i + \boldsymbol{t}^c \right), \qquad (3)$$

where $\boldsymbol{K}^c$ is the intrinsic parameter and λ_i^c is the scaling factor. For rotation estimation, when N orientations $\boldsymbol{v}_i^c$ are estimated at camera c, we obtain the following equation from equation 2.

$$\begin{aligned} \left[\begin{array}{c} \boldsymbol{v}_1^c \ldots \boldsymbol{v}_N^c \end{array} \right]^\top &= \left[\begin{array}{c} \boldsymbol{v}_1 \ldots \boldsymbol{v}_N \end{array} \right]^\top \boldsymbol{R}^{c^\top}, \\ \Leftrightarrow \quad \boldsymbol{V}^c &= \boldsymbol{V} \boldsymbol{R}^{c^\top}. && (4) \end{aligned}$$

When $\boldsymbol{v}_i^c$ are estimated over C cameras, we have

$$\begin{aligned} \left[\begin{array}{c} \boldsymbol{V}^1 \cdots \boldsymbol{V}^C \end{array} \right] &= \boldsymbol{V} \left[\begin{array}{c} \boldsymbol{R}^{1^\top} \cdots \boldsymbol{R}^{C^\top} \end{array} \right], \\ \Leftrightarrow \quad \boldsymbol{V}^{1:C} &= \boldsymbol{V} \boldsymbol{R}^{1:C}, && (5) \end{aligned}$$

where $\boldsymbol{V}^{1:C}$ is the $N \times 3C$ matrix and $\boldsymbol{R}^{1:C}$ is the $3 \times 3C$ rotation matrix. From the previous study [12], using the SVD $\boldsymbol{V}^{1:C} = \boldsymbol{Y} \boldsymbol{D} \boldsymbol{Z}^\top$ (where $\boldsymbol{Y} \in \mathbb{R}^{N \times 3}$, $\boldsymbol{D} \in \mathbb{R}^{3 \times 3}$, and $\boldsymbol{Z}^\top \in \mathbb{R}^{3 \times 3C}$), we can define $\boldsymbol{M}$ as the inverse matrix of the 3×3 submatrix on the left side of $\boldsymbol{Z}^\top$ scaled by $\sqrt{C}$ (C is the number of cameras) and factorize $\boldsymbol{V}^{1:C}$ as follows.

$$\boldsymbol{V} = \boldsymbol{Y} \boldsymbol{D} \boldsymbol{M}^{-1}, \quad \boldsymbol{R}^{1:C} = \boldsymbol{M} \boldsymbol{Z}^\top. \qquad (6)$$

From equation 6, the rotation matrix $\boldsymbol{R}^{1:C}$ is estimated using $\boldsymbol{Z}^\top$ obtained from the SVD of the matrix $\boldsymbol{V}^{1:C}$, which is the set of known estimated orientations.

After rotation estimation, collinearity and coplanarity constraints are used in [12] for translation estimation. From equations 1 and 3, $\boldsymbol{x}_i^c$ and $\boldsymbol{n}_i^c = \left[n_{i,x}^c, n_{i,y}^c, n_{i,z}^c \right]^\top = \left(\boldsymbol{K}^c \right)^{-1} \left[\boldsymbol{y}_i^c, 1 \right]^\top$ is are collinear and their cross product is zero:

$$\begin{aligned} \boldsymbol{n}_i^c \times \boldsymbol{x}_i^c &= \left[\boldsymbol{n}_i^c \right]_\times \left(\boldsymbol{R}^c \boldsymbol{x}_i + \boldsymbol{t}^c \right) \\ &= \left[\begin{array}{cc} \left[\boldsymbol{n}_i^c \right]_\times \boldsymbol{R}^c & \left[\boldsymbol{n}_i^c \right]_\times \end{array} \right] \left[\boldsymbol{x}_i \boldsymbol{t}^c \right] \\ &= \boldsymbol{0}_{3 \times 1}, && (7) \end{aligned}$$

where $\left[\boldsymbol{n}_i^c \right]_\times$ is the skew-symmetric matrix of $\boldsymbol{n}_i^c$. In addition to the collinear, since back projection through the corresponding point of cameras c and c' ($\boldsymbol{n}_i^c$ and $\boldsymbol{n}_i^{c'}$) and the vector pointing from camera c to c' ($\boldsymbol{t^c} - \boldsymbol{t^{c'}}$) are coplanar, their scalar triple product is zero:

$$\begin{aligned} &\left(\left(\boldsymbol{R}^{c\top} \boldsymbol{n}_i^c \right) \times \left(\boldsymbol{R}^{c'\top} \boldsymbol{n}_i^{c'} \right) \right)^\top \left(\boldsymbol{R}^{c\top} \boldsymbol{t}^c - \boldsymbol{R}^{c'\top} \boldsymbol{t}^{c'} \right) \\ &\quad = \left(m_i^{c,c'} \right)^\top \left(\boldsymbol{R}^{c\top} \boldsymbol{t}^c - \boldsymbol{R}^{c'\top} \boldsymbol{t}^{c'} \right) = 0 \end{aligned}, \qquad (8)$$

where $m_i^{c,c'}$ denotes $\left(\boldsymbol{R}^{c\top} \boldsymbol{n}_i^c \right) \times \left(\boldsymbol{R}^{c'\top} \boldsymbol{n}_i^{c'} \right)$. For N corresponding points, equation 8 is

$$\begin{bmatrix} \left(m_1^{c,c'} \right)^\top \boldsymbol{R}^{c\top} & -\left(m_1^{c,c'} \right)^\top \boldsymbol{R}^{c'\top} \\ \vdots & \vdots \\ \left(m_N^{c,c'} \right)^\top \boldsymbol{R}^{c\top} & -\left(m_N^{c,c'} \right)^\top \boldsymbol{R}^{c'\top} \end{bmatrix} \begin{bmatrix} \boldsymbol{t}^c \\ \boldsymbol{t}^{c'} \end{bmatrix} = \boldsymbol{0}_{N \times 1}. \qquad (9)$$

When N correspondence points are estimated from C cameras, equations 7 and 9 are the following linear equations with $3N + 3C$ unknowns $\left[\begin{array}{cccccc} \boldsymbol{x}_1 & \ldots & \boldsymbol{x}_N & \boldsymbol{t}_1 & \ldots & \boldsymbol{t}_C \end{array} \right]^\top$:

$$\boldsymbol{A} \left[\begin{array}{cccccc} \boldsymbol{x}_1 & \ldots & \boldsymbol{x}_N & \boldsymbol{t}_1 & \ldots & \boldsymbol{t}_C \end{array} \right]^\top = \boldsymbol{0}. \qquad (10)$$

$\boldsymbol{A}$ is the sparse matrix of the left side of equations 7 and 9 ($\left[\boldsymbol{n}_i^c \right]_\times \boldsymbol{R}^c$, $\left[\boldsymbol{n}_i^c \right]_\times$, $m_i^{c,c'} \boldsymbol{R}^{c\top}$, and $-m_i^{c,c'} \boldsymbol{R}^{c'\top}$), vertically arranged. We can estimate the translation $\boldsymbol{t}$ and the keypoints $\boldsymbol{x}$ in the world coordinate at the same time from equation 10. The above method also allows robust calibration against outliers using RANSAC, since extrinsic parameters can be estimated by estimating three or more corresponding key points from each camera. Lee et al. also introduce bandle adjustment for more accurate calibration [12]. However, in our research, we do not use RANSAC and bandle adjustment to avoid the high computational cost. Instead, the monocular pose estimation model is replaced with a higher performance model to maintain calibration performance (HRNet [23] and VideoPose3D [19] are replaced by ViTPose [31] and MotionAGFormer [22]).

Since camera rotation and translation can be estimated by automatic calibration, these parameters can be used to obtain 3D keypoints in the world coordinate system by triangulating from 2D keypoints. The triangulated 3D keypoints are more accurate than monocular 3D pose estimation results. Therefore, we can use the triangulated keypoints as the pseudo-labels to fine-tune the monocular 3D pose estimation model for sports motion.

2.3. Unsupervised Fine-tuning

In the fine-tuning step, we use the generated pseudo-labels and train the MotionAGFormer pre-trained with the Human3.6M dataset. Since the Human3.6M dataset does not contain the intense sports movements such as running, the pre-trained model performance is not sufficient. Therefore, there is room to improve the performance of the model, even if the pre-trained model is fine-tuned with pseudo-labels that are not accurate label data. Fine-tuning improves the performance of monocular 3D pose estimation, which in turn improves the performance of multi-view pose estimation based on automatic camera calibration. Depending on the purpose of the motion analysis and the environment, we can use either convenient monocular 3D pose estimation or multi-view pose estimation with increased accuracy.

3. Experiments

3.1. Dataset

We used the original running motion dataset and ASPset-510 [17] for experimental evaluation. Note that in both datasets, the head and nose keypoints were excluded from the evaluation because the format of the keypoints in the ground truth and in the monocular pose estimation model (Human3.6M 17 keypoints format) are slightly different.

To create the original dataset, we captured running videos from three directions in the indoor lab. Figure 3 shows the video capture environment for our dataset. The videos were captured at 4K 60fps using iPhone 11 and 13. The three runners were captured for five minutes each. Each runner ran forward and backward over a distance of approximately 8 meters. The effective capture area for all cameras to capture the runners was a rectangular area of 4 meters × 2 meters. A markerless motion capture system (Theia3D, Theia Inc.) with nine high-speed cameras (Miqus Video, Qualisys Inc.) was also used to capture the runners, and the resulting data was used as ground truth to evaluate the pose estimation.

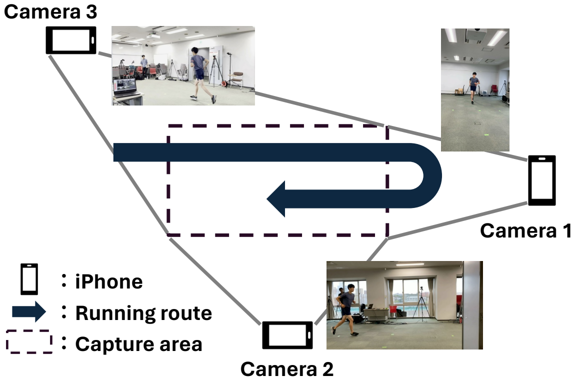


Figure 3. The video capture environment. We use smartphones to capture video. The capture area is 4 meters × 2 meters. We also capture running motion using a markerless motion capture system.

After video capture, the time of the iPhone videos was synchronized with the ground truth motion capture data, and only the time when the runner was present in all videos was trimmed. As a result, a set of videos from three directions of a runner passing through the capture area and the 3D joint position coordinates (ground truth) of the runner in the world coordinate system were obtained. In the end, a total of 233 sets, about 20,000 frames of data, were obtained. We used this dataset for unsupervised fine-tuning and to evaluate the pose estimation performance. Note that since the camera that captured the ground truth data is different from the camera that captured the video data, and there are no joint position coordinates in the camera coordinate system, the joint position coordinates in the world coordinate system are used for the evaluation.

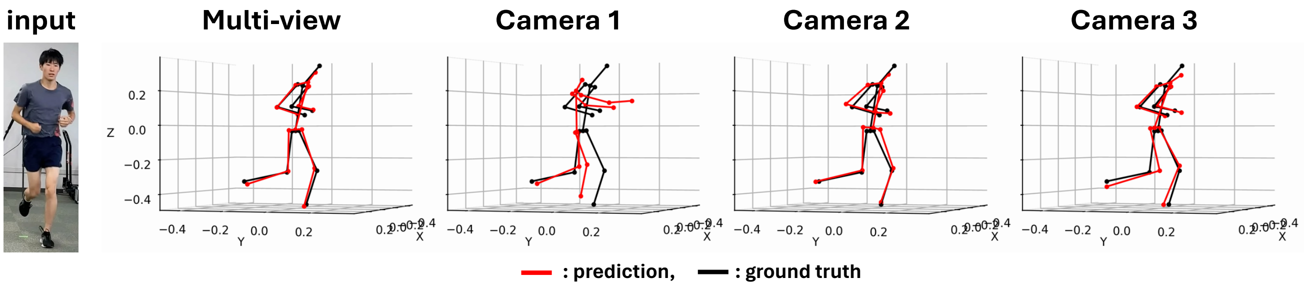
ASPset-510 contains 17 different amateur subjects performing 30 sports-related actions each, for a total of 510 action clips. Training and validation data also contains 3 videos at 4K 50fps from different directions for each action clip. Note that the test data only has video from one direction for each action clip. For the fine-tuning, we need to generate pseudo-labels using multi-view videos. Furthermore, the previous study [12] deals with a small dataset for evaluation. For example, only the *S11 Walking 1* sequence from Human3.6M was used for evaluation. Therefore, we used more data, the validation split from ASPset-510, which contains 60 action clips (about 45,000 frames), for fine-tuning and evaluation.

3.2. Implementation and Evaluation

We evaluated the performance of the monocular 3D pose estimation and multi-view 3D pose estimation based on automatic calibration before and after unsupervised fine-tuning. Our proposed system was implemented in Python 3 with Pytorch. Our code and dataset are available at https://github.com/SZucchini/unsupervised-fine-tuning-pose3d-for-sports.

For the experiment of ASPset-510, to pre-train the MotionAGFormer model using the Human3.6M dataset with scale augmentation, we prepared the same pre-processed training data as in [22, 33] and randomly rescaled the input and label from 0.1 to 0.5. We used the same parameters and loss functions for pre-training and unsupervised fine-tuning as in [22].

To evaluate monocular 3D pose estimation, we used the


input

Multi-view

Camera 1

Camera 2

Camera 3

Figure 4. Comparison of multi-view and monocular estimation results after procrustes analysis using specific examples from the running motion dataset. The red dots and line are the estimation results and the black dots and line are the ground truth pose. Camera 1 was placed in front of the runner, with large errors in both the upper and lower body. Camera 2 was placed to the right side of the runner, with large errors in the hip and elbow joints. Camera 3 was placed to the left rear side of the runner, with an error in the right knee.

Procrustes Analysis Mean Per Joint Position Error (PA-MPJPE). This metric has also been used in many previous studies [6, 19, 22, 33]. PA-MPJPE calculates the average Euclidean distance between joint positions after aligning the pose of each frame by rotation, translation, and scaling. Thus, even if the coordinate system of the predicted pose and the ground truth pose are different, the pose error can still be evaluated. In this research, since our running motion dataset has no camera coordinate ground truth data (only world coordinate), PA-MPJPE is appropriate for the evaluation metric.

However, PA-MPJPE cannot correctly evaluate the accuracy of the trajectory and distance of the pose because it aligns each frame pose by rotation and translation. Therefore, we used “Sequencial” PA-MPJPE (SPA-MPJPE) to evaluate multi-view pose estimation in addition to PA-MPJPE. In contrast to PA-MPJPE, SPA-MPJPE calculates common rotation, translation, and scaling parameters for all frames and aligns the entire pose sequence instead of aligning each frame individually. This metric also evaluates how well the trajectory matches the ground truth in the multi-view estimation results. The same idea as SPA-MPJPE was introduced as loss functions to evaluate pose consistency in another multi-view pose estimation study [6]. The units for both evaluation metrics are millimeters.

3.3. Results

In this subsection, we first present the results of the pre-training of the MotionAGFormer on the Human3.6M dataset with scale augmentation. Next, we present the results of monocular and automatic calibration-based multi-view 3D pose estimation using the MotionAGFormer pre-trained on the Human3.6M dataset. Then, we present the results of both pose estimation methods using the unsupervised fine-tuned MotionAGFormer.

Pre-training with Human3.6M with scale augmentation improved the PA-MPJPE of monocular 3D pose estimation for ASPset-510 from 207.98 (pre-training w/o scale augmentation) to 74.89. Without this pre-training, the automatic calibration using monocular 3D pose estimation results did not work well for ASPset-510. Note that we did not use ASPset-510 data for pre-training. This result indicates that the current monocular pose estimation models are affected by the scale of the training data.

Table 1. Comparison of monocular and multi-view estimation using a **pre-trained** model. Both metric units are in millimeters.

	PA-MPJPE	SPA-MPJPE
Running w/ Monocular	73.83	-
Running w/ Multi-view	55.91	123.19
ASPset w/ Monocular	74.90	-
ASPset w/ Multi-view	45.21	98.99

Table 1 shows the pose estimation results using the *pre-trained* model for each dataset. Compared to monocular pose estimation, the multi-view pose estimation based on automatic calibration performed well for both datasets. The results indicate that the automatic calibration in the previous study is also effective for sports motion. On the other hand, SPA-MPJPE was worse than PA-MPJPE, indicating that it is difficult for the results of the pre-trained model to evaluate the motion trajectory and distance moved. We also show the example pose result of the multi-view and pre-trained monocular pose estimation methods for the running motion dataset in Figure 4. In monocular estimation, the estimation error between the predicted pose and the ground truth pose was larger than in multi-view estimation. Especially for Camera 1, the error was the largest of all monocular estimation results. We assume that it is difficult for the pre-trained model to estimate a person moving in the direction of the optical axis of the camera. The above results suggest that the performance of the monocular pose estimation model could be improved by using the more accurate multi-

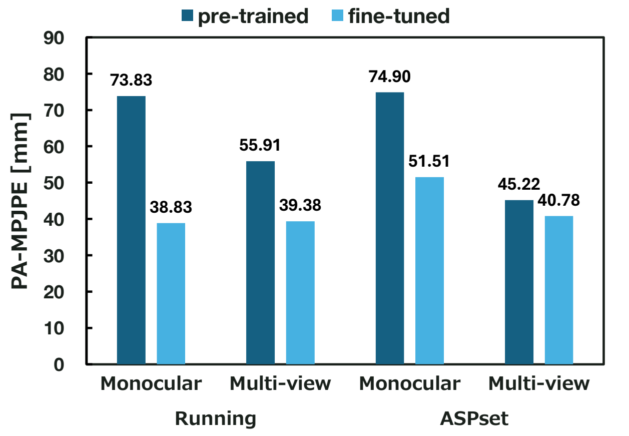


Figure 5. Comparison of PA-MPJPE of each pose estimation method before and after unsupervised fine-tuning. The fine-tuned model is the model after two iterations of fine-tuning.

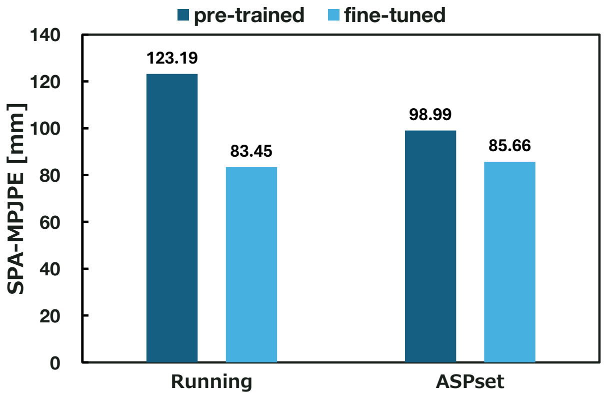


Figure 6. Comparison of SPA-MPJPE of each pose estimation method before and after unsupervised fine-tuning. SPA-MPJPE was still larger than PA-MPJPE.

view estimation results as pseudo-labels for fine-tuning.

For the fine-tuning result, we show the comparison of PA-MPJPE of each estimation method using the pre-trained and fine-tuned model in Figure 5. The fine-tuned model is the best model for PA-MPJPE during four fine-tuning iterations. The relationship between fine-tuning iterations and PA-MPJPE will be discussed later. Figure 5 shows that unsupervised fine-tuning with pseudo-labels reduces PA-MPJPE for both monocular and multi-view estimation. In particular, in a running motion dataset, the performance of monocular pose estimation was nearly equal to that of multi-view pose estimation after fine-tuning. This result indicates that we can use fine-tuned monocular 3D pose estimation alone to analyze motion that is unaffected by rotation or translation error, such as joint angles. In ASPset-510, fine-tuning also improved the performance of the monocular pose estimation model, although not as much as in the running motion dataset. The performance of the multi-view pose estimation was also improved to the same level as the running motion dataset. We also show the comparison of SPA-MPJPE in Figure 6 to evaluate the accuracy of the pose sequences. Fine-tuning also improved SPA-MPJPE, but the error was larger compared to PA-MPJPE. This suggests that the proposed method may be less effective for analyzing motion affected by rotation or translation errors, such as vertical motion or distance moved.

To evaluate the relationship between the unsupervised fine-tuning iteration and the improvement in the pose estimation results, Figure 7 shows the relationship between the fine-tuning iteration and PA-MPJPE for monocular and multi-view pose estimation. In the case of the running motion dataset, PA-MPJPE changed slightly after the second iteration for multi-view pose estimation. On the other hand, for monocular pose estimation, PA-MPJPE approached the limit of improvement after two iterations. This is because the pseudo-labels are also improved after the first iteration and contribute to the improvement of the monocular pose estimation in the second iteration. In the case of ASPset-510, PA-MPJPE approached the improvement limit for monocular and multi-view estimation after one iteration. For monocular pose estimation, the best performance of PA-

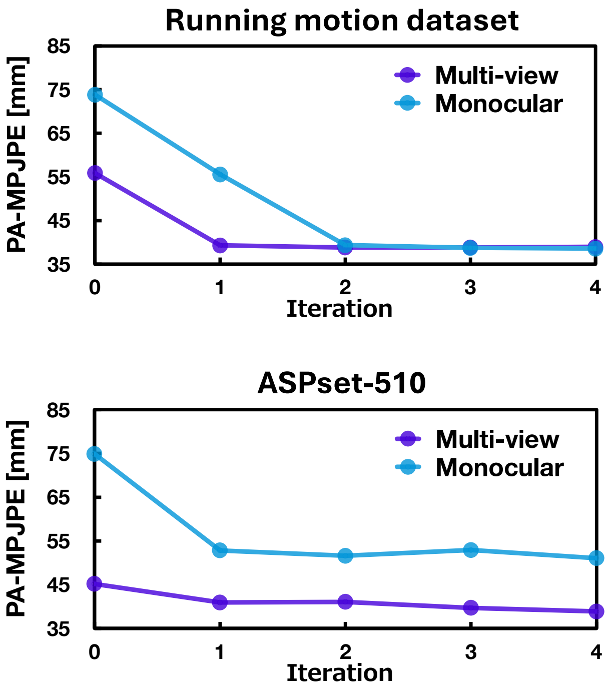


Figure 7. The relationship between the fine-tuning iteration and PA-MPJPE for monocular and multi-view pose estimation.

MPJPE in supervised learning on some datasets is around 30 millimeters [22, 33]. Although the datasets are different and cannot be strictly compared, we consider that the performance of the monocular estimation for the running motion dataset and that of the multi-view estimation for ASPset-510 after unsupervised fine-tuning was improved close to some supervised monocular models [19, 22]. In the study by Ingwersen et al. [6], although they achieve higher performance results on the sports motion dataset [7], it is based on supervised learning with constraints on camera placement and thus cannot be compared to our method.

Finally, we show the example monocular pose estimation results before and after fine-tuning for the running motion dataset in Figure 8. The pre-trained result is the same as Camera 1 result in Figure 4, and the fine-tuned result is also Camera 1. The pre-trained model had the largest estimation error for the video captured by Camera 1. However, the estimation error of Camera 1 was significantly reduced, and the result for the other cameras was the same. For the running motion dataset, fine-tuning allows for accurate monocular estimation comparable to multi-view, although it is a monocular pose estimation.

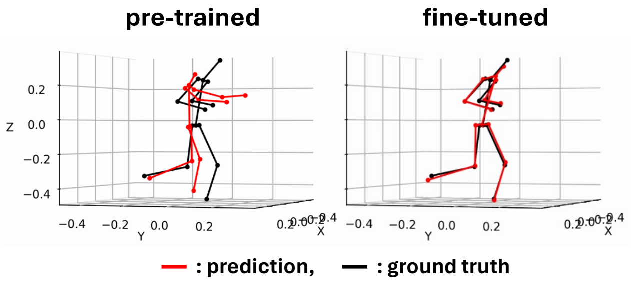


Figure 8. Comparison of monocular 3D pose estimation results of Camera 1 before and after fine-tuning using specific examples from the running motion dataset.

4. Conclusion

In this paper, we proposed a cost-effective motion capture system based on the unsupervised fine-tuned pose estimation model. For the unsupervised fine-tuning, to generate pseudo-labels, we use the automatic camera calibration method proposed in the previous study and show that it is also effective for sports motion. To evaluate our proposed system, we used the original running motion dataset and ASPset-510, which contains many kinds of sports motion videos. The evaluation results show that our method can improve the performance of the monocular 3D pose estimation model for sports motion. We also showed that the fine-tuned monocular estimation model is comparable to the multi-view estimation for the running motion. Since our method does not require special equipment, preparation for measurement, and annotation costs for model tuning, it can be useful for the daily sports training motion analysis.

While our method performed well on the data used for unsupervised fine-tuning, we could not evaluate MPJPE, which is a general metric for evaluating pose estimation, due to the limitations of our dataset. The results of the PA-MPJPE evaluation show that our method is useful for analyzing motions that are not affected by rotation or translation errors, such as joint angles. However, for more useful motion analysis, we need to evaluate MPJPE in future work. In addition, the versatility of our method for other motion data is unclear. The performance also differed between datasets for the running motion and ASPset-510. To improve our method, defining the confidence score of pseudo-labels by spatio-temporal constraints on motion and weighting labels during training could be effective.

Acknowledgements. We would like to thank Prof. Nobuhara of Kyoto Institute of Technology for discussions on camera calibration techniques. This work was financially supported by JST PRESTO Grant Number JPMJPR20CA.

References

[1] 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. 1

[2] Bowen Cheng, Bin Xiao, Jingdong Wang, Honghui Shi, Thomas S Huang, and Lei Zhang. Higherhrnet: Scale-aware representation learning for bottom-up human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5386–5395, 2020. 1

[3] Hai Ci, Xiaoxuan Ma, Chunyu Wang, and Yizhou Wang. Locally connected network for monocular 3d human pose estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(3):1429–1442, 2020. 1, 2

[4] Jon Echeverria and Olga C Santos. Toward modeling psychomotor performance in karate combats using computer vision pose estimation. *Sensors (Basel)*, 21(24):8378, 2021. 1

[5] Rıza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1

[6] Christian Keilstrup Ingwersen, Anders Bjorholm Dahl, Janus Nørtoft Jensen, and Morten Rieger Hannemose. Two views are better than one: Monocular 3d pose estimation with multiview consistency. *arXiv preprint arXiv:2311.12421*, 2023. 1, 2, 6, 8

[7] Christian Keilstrup Ingwersen, Christian Mikkelstrup, Janus Nørtoft Jensen, Morten Rieger Hannemose, and Anders Bjorholm Dahl. Sportspose: A dynamic 3d sports pose dataset. In *Proceedings of the IEEE/CVF International Workshop on Computer Vision in Sports*, 2023. 3, 8

[8] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian 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, 2013. 2, 3

[9] Hanbyul Joo, Hao Liu, Lei Tan, Lin Gui, Bart Nabbe, Iain Matthews, Takeo Kanade, Shohei Nobuhara, and Yaser Sheikh. Panoptic studio: A massively multiview system for social motion capture. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3334–3342, 2015. 2

[10] Aftab Khan, Sebastian Mellor, Rachel King, Balazs Janko, William Harwin, R. Simon Sherratt, Ian Craddock, and Thomas Plötz. Generalized and efficient skill assessment from imu data with applications in gymnastics and medical training. *ACM Trans. Comput. Healthcare*, 2(1), 2021. 1

[11] Muhammed Kocabas, Nikos Athanasiou, and Michael J. Black. Vibe: Video inference for human body pose and shape estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[12] Sang-Eun Lee, Keisuke Shibata, Soma Nonaka, Shohei Nobuhara, and Ko Nishino. Extrinsic camera calibration from a moving person. *IEEE Robotics and Automation Letters*, 7(4):10344–10351, 2022. 2, 3, 4, 5

[13] Wenhao Li, Hong Liu, Runwei Ding, Mengyuan Liu, Pichao Wang, and Wenming Yang. Exploiting temporal contexts with strided transformer for 3d human pose estimation. *IEEE Transactions on Multimedia*, 25:1282–1293, 2023. 1, 2

[14] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 2, 3

[15] Julieta Martinez, Rayat Hossain, Javier Romero, and James J. Little. A simple yet effective baseline for 3d human pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017. 1

[16] Gaku Nakano. Camera calibration using parallel line segments. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 1505–1512. IEEE, 2021. 2

[17] Aiden Nibali, Joshua Millward, Zhen He, and Stuart Morgan. ASPset: An outdoor sports pose video dataset with 3D keypoint annotations. *Image and Vision Computing*, page 104196, 2021. 2, 3, 5

[18] Irzan Fajari Nurahmadan, Jayanta, and I Wayan Widi Pradnyana. Utilization of pose estimation and multilayer perceptron methods in the development of taekwondo martial arts independent learning. *2021 International Conference on Informatics, Multimedia, Cyber and Information System (ICIMCIS*, pages 267–272, 2021. 1

[19] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1, 2, 4, 6, 8

[20] Jens Puwein, Luca Ballan, Remo Ziegler, and Marc Pollefeys. Joint camera pose estimation and 3d human pose estimation in a multi-camera setup. In *Computer Vision–ACCV 2014: 12th Asian Conference on Computer Vision, Singapore, Singapore, November 1-5, 2014, Revised Selected Papers, Part II 12*, pages 473–487. Springer, 2015. 2

[21] Edoardo Remelli, Shangchen Han, Sina Honari, Pascal Fua, and Robert Wang. Lightweight multi-view 3d pose estimation through camera-disentangled representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6040–6049, 2020. 1, 2

[22] Babak Taati Soroush Mehraban, Vida Adeli. Motionagformer: Enhancing 3d human pose estimation with a transformer-gcnformer network. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024. 1, 2, 3, 4, 5, 6, 8

[23] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5693–5703, 2019. 1, 4

[24] Tomohiro Suzuki, Kazuya Takeda, and Keisuke Fujii. Automatic fault detection in race walking from a smartphone camera via fine-tuning pose estimation. In *2022 IEEE 11th Global Conference on Consumer Electronics (GCCE)*, pages 631–632. IEEE, 2022. 1, 2

[25] Tomohiro Suzuki, Kazuya Takeda, and Keisuke Fujii. Automatic detection of faults in simulated race walking from a fixed smartphone camera. *International Journal of Computer Science in Sport*, 23(1):22–36, 2024. 1, 2

[26] Kosuke Takahashi, Dan Mikami, Mariko Isogawa, and Hideaki Kimata. Human pose as calibration pattern; 3d human pose estimation with multiple unsynchronized and uncalibrated cameras. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1775–1782, 2018. 2

[27] Ryota Tanaka, Tomohiro Suzuki, Kazuya Takeda, and Keisuke Fujii. Automatic edge error judgment in figure skating using 3d pose estimation from inertial sensors. In *2023 IEEE 12th Global Conference on Consumer Electronics (GCCE)*, pages 1099–1100. IEEE, 2023. 1

[28] Ryota Tanaka, Tomohiro Suzuki, Kazuya Takeda, and Keisuke Fujii. Automatic edge error judgment in figure skating using 3d pose estimation from a monocular camera and imus. In *Proceedings of the 6th International Workshop on Multimedia Content Analysis in Sports*, pages 41–48, 2023. 1

[29] Scott D Uhlrich, Antoine Falisse, Łukasz Kidziński, Julie Muccini, Michael Ko, Akshay S Chaudhari, Jennifer L Hicks, and Scott L Delp. Opencap: Human movement dynamics from smartphone videos. *PLoS computational biology*, 19(10):e1011462, 2023. 1, 2

[30] Timo Von Marcard, Roberto Henschel, Michael J Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *Proceedings of the European conference on computer vision (ECCV)*, pages 601–617, 2018. 2

[31] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. Vitpose: Simple vision transformer baselines for human pose estimation. *Advances in Neural Information Processing Systems*, 35:38571–38584, 2022. 1, 3, 4

[32] Xinyu Yi, Yuxiao Zhou, Marc Habermann, Soshi Shimada, Vladislav Golyanik, Christian Theobalt, and Feng 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*, pages 13167–13178, 2022. 1

[33] Wentao Zhu, Xiaoxuan Ma, Zhaoyang Liu, Libin Liu, Wayne Wu, and Yizhou Wang. Motionbert: A unified perspective on learning human motion representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023. 1, 2, 5, 6, 8