# WHAM: Reconstructing World-grounded Humans with Accurate 3D Motion

Soyong Shin[1] Juyong Kim[1] Eni Halilaj[1] Michael J. Black[2]

[1]Carnegie Mellon University [2]Max Planck Institute for Intelligent System

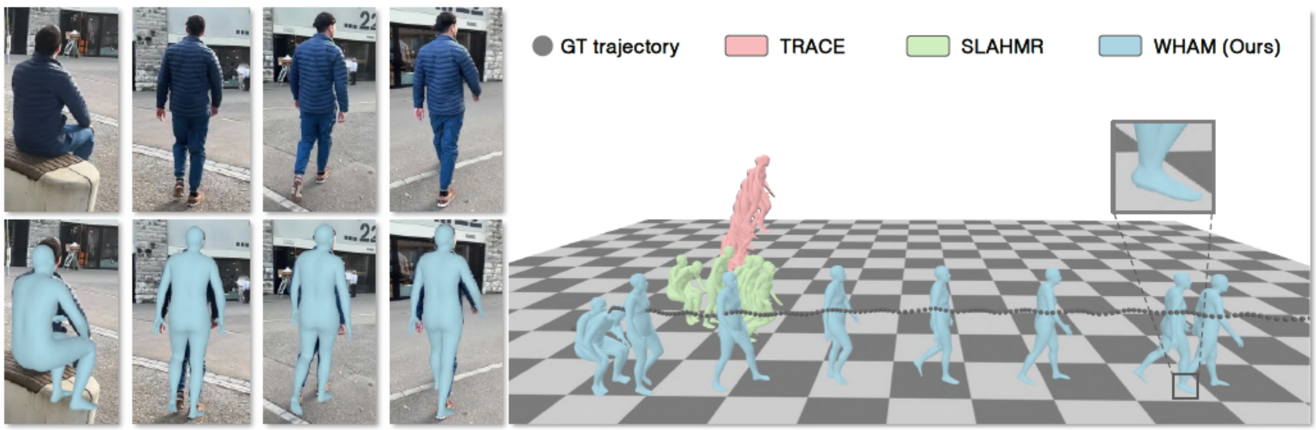


Figure 1. **WHAM: World-grounded Humans with Accurate Motion.** State-of-the-art methods like TRACE [45] and SLAHMR [54] fail to capture global 3D human trajectories accurately when given in-the-wild videos captured using a moving camera, producing implausible world-grounded motion (e.g., foot sliding). To address this, WHAM uses two novel strategies: (1) feature integration from 2D keypoints and pixels to reconstruct precise and pixel-aligned 3D human motion and (2) contact-aware trajectory recovery to place the human in a global coordinate system without foot sliding. Gray dots show the ground-truth global trajectory. See **Supplemental Video.**

## Abstract

*The estimation of 3D human motion from video has progressed rapidly but current methods still have several key limitations. First, most methods estimate the human in camera coordinates. Second, prior work on estimating humans in global coordinates often assumes a flat ground plane and produces foot sliding. Third, the most accurate methods rely on computationally expensive optimization pipelines, limiting their use to offline applications. Finally, existing video-based methods are surprisingly less accurate than single-frame methods. We address these limitations with WHAM (World-grounded Humans with Accurate Motion), which accurately and efficiently reconstructs 3D human motion in a global coordinate system from video. WHAM learns to lift 2D keypoint sequences to 3D using motion capture data and fuses this with video features, integrating motion context and visual information. WHAM exploits camera angular velocity estimated from a SLAM method together with human motion to estimate the body's global trajectory. We combine this with a contact-aware trajectory refinement method that lets WHAM capture human motion in diverse conditions, such as climbing stairs. WHAM outperforms all existing 3D human motion recovery methods across multiple in-the-wild benchmarks. Code will be available for research purposes at http://wham.is.tue.mpg.de/.*

## 1. Introduction

Our goal is to accurately estimate the 3D pose and shape of a person from monocular video. This is a longstanding problem and, while the field has made rapid progress, several key challenges remain. First, human motion should be computed in a consistent global coordinate system. Second, the method should be computationally efficient, supporting real-time processing. Third, the results should be accurate, temporally smooth, detailed, natural looking, and have realistic foot-ground contact. Fourth, the capture should work with an arbitrary moving camera. These constraints need to be satisfied to make markerless human motion capture widely available for applications in gaming, AR/VR, autonomous driving, sports analysis, and human-robot interaction. We address these challenges with *WHAM (World-grounded Humans with Accurate Motion)*, which enables fast and accurate recovery of 3D human motion from a moving camera; see Fig. 1.

It seems natural that, in estimating 3D humans from video, we should be able to exploit the temporal nature of video. Counter-intuitively, existing video-based methods for 3D human pose and shape (HPS) estimation [6, 15, 17, 30, 43, 51] are less accurate than the best single-frame methods [7, 14, 18, 21, 23, 25, 59, 59]. This may be an issue of training data. There are large datasets of single images with ground-truth 3D human poses containing a diversity of body shapes, poses, backgrounds, lighting, etc. In contrast, video datasets with ground truth are much more limited.

To address this, WHAM leverages both the large-scale AMASS motion capture (mocap) dataset [32] and video datasets. Our key idea is to learn about 3D human motion from AMASS and then learn to fuse this information with temporal image cues from video, getting the best of both. Similar to previous work [60, 61], we use AMASS to generate synthetic 2D keypoints and ground-truth motion sequence pairs, from which we pretrain a motion encoder and decoder to *lift* sequences of 2D keypoints to sequences of 3D poses. This motion encoder captures the essential motion context recursively over time, while the decoder translates this context into realistic 3D motion. Given the robustness of recent 2D keypoint detection models [53, 58], our pretrained model does a good job of predicting human pose from video.

Keypoints alone, however, are too sparse for accurate, unambiguous, 3D mesh estimation. To improve accuracy, we jointly train a new feature integrator network that merges information from video sequences and 2D-keypoint sequences. Like the motion context, the image features are integrated over time and the image encoder and feature integrator are trained using video datasets with known 3D pose and shape [11, 15, 33, 49]. This integration process supplements the motion context extracted from the sparse input signal (i.e., 2D keypoints) with dense visual context, significantly improving the recovered pose and shape accuracy.

While the above approach produces accurate motion, we want this motion in a global coordinate. Most previous methods, compute the body in camera coordinates. Estimating the global human trajectory is particularly challenging when the camera is moving because the motions of the body and the camera are entangled. Recent work addresses this challenge by exploiting a learned prior distribution over human motions, together with camera information from SLAM methods [20, 41, 54] or dense 3D scene information from COLMAP [28], to solve for global human motion using optimization. However, these optimization-based methods are computationally expensive and far from real time. Recent regression-based methods are faster but either constrain the problem with static or known camera conditions [42, 60] or have temporal jitter and limited accuracy [45]. We tackle this challenge with two additional modules. First, we predict the global orientation and egocentric velocity of the human from the motion context by training a *global trajectory decoder*. Specifically, we concatenate the camera's angular velocity to the context and train the global trajectory decoder to recursively predict the current orientation and root velocity, effectively factoring camera motion from human motion. Global translation is computed by roll-out. WHAM takes the camera's angular velocity either from the output of a SLAM method or from a camera's gyroscope when available. We demonstrate the use of both.

The above solution relies on knowledge of human motion learned from AMASS. While this works well when people are moving on a flat ground plane, it can fail to capture elevation changes when the surface is not flat, e.g. when ascending the stairs. Even though we have no explicit ground-plane assumption (unlike [54]), this occurs because AMASS has a limited amount of such data. To address this, we introduce foot contact as an additional explicit form of motion context. We train WHAM to predict the likelihood of foot-ground contact using estimated contact labels from both the AMASS and 3D video datasets. We then train a trajectory refinement network that outputs an update to the root orientation and velocity based on the information about the foot contact/velocity. This refinement based on foot contact enables WHAM to accurately estimate human motion in a global coordinate system even when the terrain is not flat.

WHAM has very low computational overhead because it is an on-line algorithm that recursively predicts the pose, shape, and global motion parameters. The network, excluding preprocessing (bounding box detection, keypoint detection, and person identification), runs at 200 fps, significantly faster than prior methods. Also, despite not using global optimization like [54], we obtain accurate 3D camera trajectories and global body motions with minimal drift. Through extensive comparisons on multiple in-the-wild datasets as well as detailed ablation studies, we find that WHAM achieves state-of-the-art (SOTA) accuracy on 3D human pose estimation as well as global trajectory estimation (see Fig. 1).

In summary, in this paper we: (1) introduce the first approach to effectively fuse 3D human motion context and video context for 3D HPS regression; (2) propose a novel global trajectory estimation framework that leverages motion context and foot contact to effectively address foot sliding and enable the 3D tracking of people on non-planar surfaces; (3) perform HPS in global coordinates efficiently; (4) achieve state-of-the-art (SOTA) performance on multiple in-the-wild benchmark datasets (3DPW [49], RICH [10], EMDB [16]). WHAM is the first video-based method to outperform all image-based and video-based methods on per-frame accuracy (MPJPE, PA-MPJPE, and PVE) while maintaining temporal smoothness (Accel). Pretrained models and training code will be available for research purposes.

## 2. Related Work

**Image-based 3D HPS Estimation.** There are two broad classes of methods for recovering 3D HPS from images: model-free [22, 27, 34] and model-based [7, 12–14, 21, 25, 35, 36]. Here we focus on model-based methods, which estimate the low-dimensional parameters of a statistical body model [29, 37, 38, 52]. While early work explores optimization-based methods [2], here we focus on direct regression methods based on deep learning.

Many existing methods follow the architecture of HMR [14], which uses a pre-trained backbone to predict image features followed by a multilayer perceptron (MLP) that regresses SMPL [29] pose parameters from image features. To train such networks it leverages paired images with SMPL parameters, which are often pseudo-groundtruth SMPL parameters estimated from 2D keypoints [12, 21, 25, 35, 36]. Other architectures for HPS regression have also been proposed [7, 18, 19, 23, 24, 59]. None of these methods use video or estimate the body in global coordinates. While quite accurate, when these image-based models are applied independently to frames of a video sequence, the shape and pose can be temporally inconsistent. In contrast, WHAM effectively aggregates temporal information to provide frame-accurate and temporally-coherent 3D HPS estimation.

**Video-Based 3D HPS.** Video-based methods typically encode temporal information by combining static features extracted by a backbone from each frame. HMMR [15] uses a convolutional encoder, while VIBE [17] and MEVA [30] employ recurrent neural networks. TCMR [6] divides sequences into past, future, and whole frames, aggregating information to strongly constrain the output with motion consistency. MPS-Net [51] uses attention to capture non-local motion context and a hierarchical architecture to aggregate temporal features. Both MAED [50] and GLoT [43] use transformer architectures [48] to encode videos. MAED encodes videos in both temporal (across frames) and spatial (within each frame) dimensions and leverages the kinematic tree to iteratively regress each joint angle. GLoT encodes long-term temporal correlations and refines local details by focusing on nearby frames. Despite integrating information across frames, all existing video-based methods have lower accuracy than the best single-frame methods.

Given limited video training data with ground truth SMPL poses, several single-frame methods infer a mesh from 2D/3D keypoints [5, 8, 31, 34, 39] and use the keypoints as a proxy for training. Another set of approaches exploits 3D mocap data, which is plentiful [32], to train a network to lift 2D joints to 3D poses, which are used as a proxy for ground truth 3D. MotionBERT [61] synthesizes 2D keypoints through orthographic projection to learn unified motion representation. ProxyCap [60] creates virtual cameras with heuristic pose distribution, on which synthetic 3D keypoints are projected. Despite benefiting from the scale of the mocap dataset, these approaches do not fully utilize the visual information available in the video at test time. In our work, we propose a combined network architecture and training strategy that leverages both proxy representations of human pose (lifting) and visual context extracted from video.

**Global 3D Motion Estimation with Dense Sensors.** Several methods augment video data with other sensors to estimate 3D HPS in world coordinates. The 3DPW dataset [49] employs pre-calibrated body-worn inertial sensors and a handheld camera to jointly optimize the camera and human motion in challenging environments. Similarly, the EMDB dataset [16] uses electromagnetic sensors with an RGB-D camera, enabling accurate human motion capture in the world. Although body-worn sensors aid reconstruction of global human motion, they are intrusive, require cooperation, and do not help with archival video. BodySLAM++ [9] introduces a rapid optimization method using a visual-inertial sensor, comprising stereo cameras and an IMU. In contrast, we use a standard monocular camera, balancing accessibility and accuracy without the need for specialized equipment. While WHAM can take the camera gyro as input, this is not required.

**Monocular Global 3D Human Trajectory Estimation.** Estimating the global human trajectory from a monocular dynamic camera is challenging. Previous work relies on learned prior distributions of human motion to separate human motion from camera motion. GLAMR [57] predicts the global trajectory based on a predicted and infilled 3D motion sequence and optimizes it across multiple individuals in the scene. However, since GLAMR does not consider camera motion cues, the output trajectory may be noisy when the camera is rotating. SLAHMR [54] and PACE [20] use off-the-shelf SLAM algorithms [46, 47] and jointly optimize the camera and human motion to minimize the negative log likelihood of a learned motion prior [40]. While they achieve good results, their optimization approach is computationally expensive. TRACE [45] is a pure regression method that utilizes optical flow as a motion cue and estimates multiple people at once, but lacks temporal consistency. GloPro [42] regresses the uncertainty of the global human motion in real time, but requires known camera poses. In contrast, WHAM leverages both explicit and implicit prior knowledge of human motion and efficiently reconstructs accurate and temporally coherent 3D human motion in world coordinates.

## 3. Methods

### 3.1. Overview

An overview of our World-grounded Human with Accurate Motion (WHAM) framework is illustrated in Fig. 2. The input to WHAM is a raw video data $\{I^{(t)}\}_{t=0}^{T}$, captured by a

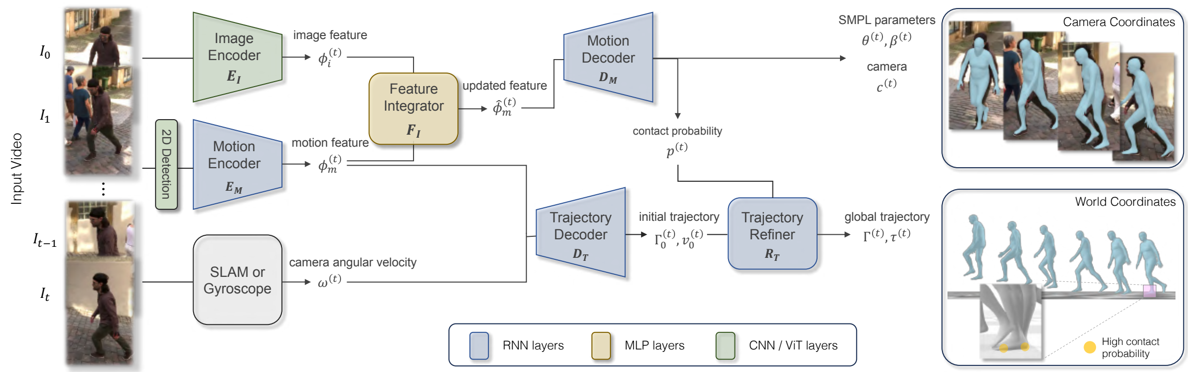


Figure 2. **An Overview of WHAM.** WHAM takes the sequence of 2D keypoints estimated by a pretrained detector and encodes it to the motion feature. WHAM then updates the motion feature using another sequence of image features extracted from the image encoder through the feature integrator. From the updated motion feature, the Local Motion Decoder estimates 3D motion in the camera coordinate system and foot-ground contact probability. The Trajectory Decoder takes the motion feature and camera angular velocity to initially estimate the global root orientation and egocentric velocity, which are then updated through the Trajectory Refiner using the foot-ground contact. The final output of WHAM is pixel-aligned 3D human motion with the 3D trajectory in the global coordinates.

camera with possibly unknown motion. Our goal is to predict the corresponding sequence of the SMPL model parameters $\{\Theta^{(t)}\}_{t=0}^{T}$, as well as the root orientation $\{\Gamma^{(t)}\}_{t=0}^{T}$ and translation $\{\tau^{(t)}\}_{t=0}^{T}$, expressed in the world coordinate system. We use ViTPose [53] to detect 2D keypoints $\{x_{2D}^{(t)}\}_{t=0}^{T}$ from which we obtain motion features $\{\phi_m^{(t)}\}_{t=0}^{T}$ using the motion encoder. Additionally, we use a pretrained image encoder [7, 21, 25] to extract static image features $\{\phi_i^{(t)}\}_{t=0}^{T}$ and integrate them with $\{\phi_m^{(t)}\}_{t=0}^{T}$ to obtain fine-grained motion features $\{\hat{\phi}_m^{(t)}\}_{t=0}^{T}$ from which we regress 3D human motion in the world coordinate system.

### 3.2. Network Architecture

**Uni-directional Motion Encoder and Decoder.** In contrast to existing methods [6, 30, 43, 51, 61], which use windows with a fixed time duration, we use uni-directional recurrent neural networks (RNN) for the motion encoder and decoder, making WHAM suitable for online inference. The objective of the motion encoder $E_M$ is to extract the motion context $\phi_m^{(t)}$ from the current and previous sequence of 2D keypoints and the initial hidden state $h_E^{(0)}$:

$$\phi_m^{(t)} = E_M\left(x_{2D}^{(0)}, x_{2D}^{(1)}, ..., x_{2D}^{(t)} | h_E^{(0)}\right).$$

We normalize keypoints to a bounding box around the person and concatenate the box's center and scale to the keypoints, similar to CLIFF [25]. The role of the motion decoder $D_M$ is to recover SMPL parameters $(\theta, \beta)$, weak-perspective camera translation $c$, and foot-ground contact probability $p$, from the motion feature history:

$$\left(\theta^{(t)}, \beta^{(t)}, c^{(t)}, p^{(t)}\right) = D_M\left(\hat{\phi}_m^{(0)}, ..., \hat{\phi}_m^{(t)} | h_D^{(0)}\right).$$

Here, $\hat{\phi}_m^{(t)}$ is the motion feature integrated with the image feature $\phi_i^{(t)}$ (described below). During pre-training on synthetic data, the image feature is not available and we set $\hat{\phi}_m^{(t)} = \phi_m^{(t)}$. As the encoder and decoder are tasked with reconstructing a dense 3D representation $\Theta$ from a sparse 2D input signal $x_{2D}$, we design an intermediate task to predict the 3D keypoints $x_{3D}$ and use them as the intermediate motion representation. This cascaded approach guides $\phi_m$ to represent the implicit context of motion and the 3D spatial structure of the body. We initialize the hidden states of the motion encoder and decoder, $\left(h_E^{(0)}, h_D^{(0)}\right)$, following PIP [56]; See Sup. Mat. for details.

**Motion and Visual Feature Integrator.** We use the AMASS dataset to synthetically generate 2D sequences by projecting 3D SMPL joints into images with varied camera motions. This provides effectively limitless training data that is far more diverse than existing video datasets that contain ground truth 3D pose and shape. Although we leverage the temporal human motion context, lifting 2D keypoints to 3D meshes is an ambiguous task. A key idea is to augment this 2D keypoint information with image cues that can help disambiguate the 3D pose. Specifically, we use an image encoder [1, 7, 21, 25], pretrained on the human mesh recovery task, to extract image features $\phi_i$, which contain dense visual contextual information related to the 3D human pose and shape. We then train a feature integrator network, $F_I$, to combine $\phi_m$ with $\phi_i$, integrating motion and visual context. The feature integrator uses a simple yet effective residual connection:

$$\hat{\phi}_m^{(t)} = \phi_m^{(t)} + F_I\left(\text{concat}\left(\phi_m^{(t)}, \phi_i^{(t)}\right)\right).$$

This supplements motion features pre-trained on the 2D-to-3D lifting task using AMASS with visual context, resulting in enriched motion features that use image evidence to help disambiguate the task.

**Global Trajectory Decoder.** We design an additional decoder, $D_T$, to predict the rough global root orientation $\Gamma_0^{(t)}$ and root velocity $v_0^{(t)}$ from the motion feature $\phi_m^{(t)}$. Since $\phi_m$ is derived from the input signals in the camera coordinates, it is highly challenging to decouple the human and camera motion from it. To address this ambiguity, we append the angular velocity of the camera, $\omega^{(t)}$, to the motion feature, $\phi_m^{(t)}$, to create a camera-agnostic motion context. This design choice makes WHAM compatible with both off-the-shelf SLAM algorithms [46, 47] and gyroscope measurements that are widely available from modern digital cameras. We recursively predict global orientation, $\Gamma_0^{(t)}$, using the uni-directional RNN.Similar to GLAMR [57], we use the ego-centric root velocity to make the prediction invariant to the global orientation:

$$\left(\Gamma_0^{(t)}, v_0^{(t)}\right) = D_T\left(\phi_m^{(0)}, \omega^{(0)}, ..., \phi_m^{(t)}, \omega^{(t)}\right).$$

**Contact Aware Trajectory Refinement.** Good 3D motion in world coordinates in most scenarios implies accurate foot-ground contact without sliding. We want WHAM to generalize beyond flat ground planes, which are typically assumed in prior work. Specifically, our new trajectory refiner aims to resolve foot sliding and enables WHAM to generalize well to diverse motions, including climbing stairs. The refinement involves two stages. First, we adjust the ego-centric root velocity to $\tilde{v}^{(t)}$ to minimize foot sliding, based on the foot-ground contact probability $p^{(t)}$:

$$\tilde{v}^{(t)} = v_0^{(t)} - \left(\Gamma_0^{(t)}\right)^{-1} \bar{v}_f^{(t)},$$

where $\bar{v}_f^{(t)}$ is the averaged velocity of the toes and heels in the world coordinate when their contact probability, $p^{(t)}$, is higher than a threshold. However, this velocity adjustment often introduces noisy translation when the contact and pose estimation is inaccurate. Therefore, we propose a simple learning mechanism in which a trajectory refining network, $R_T$, updates the root orientation and velocity to address this issue. Finally, the global translation is computed through a roll-out operation:

$$\begin{aligned} \left(\Gamma^{(t)}, v^{(t)}\right) &= R_T\left(\phi_m^{(0)}, \Gamma_0^{(0)}, \tilde{v}^{(0)}, ..., \phi_m^{(t)}, \Gamma_0^{(t)}, \tilde{v}^{(t)}\right), \\ \tau^{(t)} &= \sum_{i=0}^{t-1} \Gamma^{(i)} v^{(i)}. \end{aligned}$$

In summary, this full process reconstructs accurate 3D human pose and shape in both the camera and world coordinates from a single monocular video sequence (Fig. 2).

### 3.3. Training

**Pretraining on AMASS.** We train in two stages: (1) pretraining with synthetic data, and (2) fine-tuning with real data (Fig. 3). The objective of the pretraining stage is to teach the motion encoder to extract motion context from the input 2D keypoint sequence. The motion and trajectory decoders then map this motion context to the corresponding 3D motion and global trajectory spaces (i.e. they lift the encoding to 3D). We use the AMASS dataset [32] to generate an extensive set of synthetic pairs consisting of sequences of 2D keypoints together with the ground truth SMPL parameters.

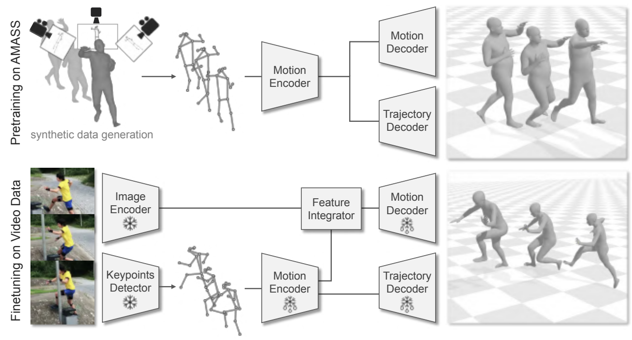


Figure 3. **WHAM's Two-Stage Training Scheme.** During pretaining, we generate synthetic 2D keypoint sequences from AMASS [32] and train a motion encoder and decoder on the generated data (*top*). We then leverage video datasets with ground truth SMPL parameters, for which there is much less data. We use the fixed-weight pre-trained image encoder and keypoints detector (❄) to extract image features and 2D keypoints. In this stage, we train the feature integration network while fine-tuning the motion encoder and motion/trajectory decoders, marked ❄ (*bottom*).

To synthesize 2D keypoints from AMASS, we create virtual cameras onto which we project 3D keypoints derived from the ground truth mesh. Unlike MotionBERT [61] and ProxyCap [60], which use static cameras for keypoint projection, we employ dynamic cameras that incorporate both rotational and translational motion. This choice is based on two main reasons. First, it accounts for the inherent differences between human motion captured in static and dynamic camera setups. Second, it enables the learning of a camera-agnostic motion representation, from which the trajectory decoder can reconstruct the global trajectory. We also augment the 2D data with noise and masking. For details of the synthetic generation process see Sup. Mat.

**Fine-tuning on Video Datasets.** Starting with the pre-trained motion encoder and decoders, we fine-tune WHAM on four video datasets: 3DPW [49], Human3.6M [11], MPI-INF-3DHP [33], and InstaVariety [15]. For the human mesh recovery task, we supervise WHAM on ground-truth SMPL parameters from AMASS and 3DPW, 3D keypoints from Human3.6M and MPII3D, and 2D keypoints from InstaVariety. For the global trajectory estimation task, we use AMASS, Human3.6M, and MPII3D. Additionally, during training we experiment with adding BEDLAM [1](which we call WHAM-B), a large synthetic dataset with realistic video and ground truth SMPL parameters.

The fine-tuning has two objectives: 1) exposing the network to real 2D keypoints, instead of training it solely on synthetic data, and 2) training the feature integrator network to aggregate motion and image features. To achieve these goals, we jointly train the entire network on the video datasets while setting a smaller learning rate on the pre-trained modules (see Fig. 3). Consistent with prior work [6, 17, 30, 43, 51], we employ a pre-trained and fixed-weight image encoder [21] to extract image features. However, to leverage recent network architectures and training strategies, we also experiment with different types of encoders [1, 7, 25] in the following section.

## 4. Experiments

**Datasets.** We evaluate WHAM on three in-the-wild benchmarks: 3DPW [49], RICH [10], and EMDB [16]. Following previous work [1, 6, 14, 17, 21, 25, 30], we perform the evaluation in camera coordinates. The estimated global trajectory is evaluated on a subset of EMDB (*EMDB 2*) for which they provide ground truth global motion with dynamic cameras (used for evaluation). We also test on new sequences captured using an iPhone with ground-truth camera angular rotation from the gyroscope. See Sup. Mat. for more details of the datasets and iPhone results.

**Evaluation metrics.** To evaluate the accuracy of 3D human pose and shape estimation, we compute Mean Per Joint Position Error (MPJPE), Procrustes-aligned MPJPE (PA-MPJPE), and Per Vertex Error (PVE) measured in millimeters $(mm)$. We compute Acceleration error (Accel, in $m/s^2$)[1] to measure the inter-frame smoothness of the reconstructed motion. We also evaluate the motion reconstruction and trajectory estimation accuracy in the world-frame. Following previous work [20, 54], we split sequences into smaller segments of 100 frames and align each output segment with the ground-truth data using the first two frames (W-MPJPE$_{100}$) or the entire segment (WA-MPJPE$_{100}$) in $mm$. These previous metrics give an unrealistic picture of 3D performance as they do not measure drift over long sequences. Therefore, we also evaluate the error over the *entire trajectory* after aligning with the initial camera pose and measure the Root Orientation Error (ROE in $\deg$) and Root Translation Error (RTE in $m$). We also assess the consistency of human motion in the global coordinate system using the Ego-centric Root Velocity Error (ERVE) and Ego-centric Foot Velocity Error (EFVE) in $mm$/frame.

### 4.1. 3D Human Motion Recovery

**Per-frame accuracy.** In Table 1, we present a comprehensive comparison of WHAM and the existing state-of-the-art per-frame and video-based methods across three benchmark datasets [10, 16, 49]. WHAM (Res), WHAM (HR), and WHAM (ViT) correspond to different architectures for the pre-trained image encoders, derived from SPIN (ResNet-50) [21], CLIFF (HRNet-W48) [1, 25], and HMR2.0 (ViT-H/16) [7], respectively. Not surprisingly, WHAM (HR) is more accurate than WHAM (Res), while the transformer-based version, WHAM (ViT), is the most accurate. The backbone matters, with WHAM (ViT) outperforming all previous methods on all per-frame metrics (MPJPE, PA-MPJPE, and PVE) on all benchmarks. Because none of the methods are exposed to data from RICH or EMDB during training, results on these datasets are indicative of each method's ability to generalize. Even with the simplest ResNet backbone, WHAM (Res) outperforms every method with the exception of BEDLAM-CLIFF on RICH. Unlike BEDLAM-CLIFF, WHAM (Res) is not trained on the BEDLAM dataset. Note that training on BEDLAM consistently improves accuracy in prior work (see [1, 3]), and we find the same here when we add BEDLAM to the training data (WHAM-B). SLAHMR results for 3DPW and RICH are from [54], while EMDB results are computed with their released code.

**Inter-frame smoothness.** We also evaluate the inter-frame smoothness using the acceleration error. Compared with state-of-the-art per-frame methods [1, 7, 23, 25], WHAM has significantly lower acceleration error. This indicates that WHAM reconstructs smooth and more plausible 3D human motion across frames while not sacrificing high per-frame accuracy. Conversely, when compared to recent temporal methods [6, 43, 51], WHAM exhibits marginally higher acceleration error. However, we observe that these video-based methods tend to over-smooth the human motion, resulting in lower accuracy on per-frame metrics.

To provide intuition for these numbers, we qualitatively compare WHAM with TCMR [6] and GLoT [43] in Fig. 4.

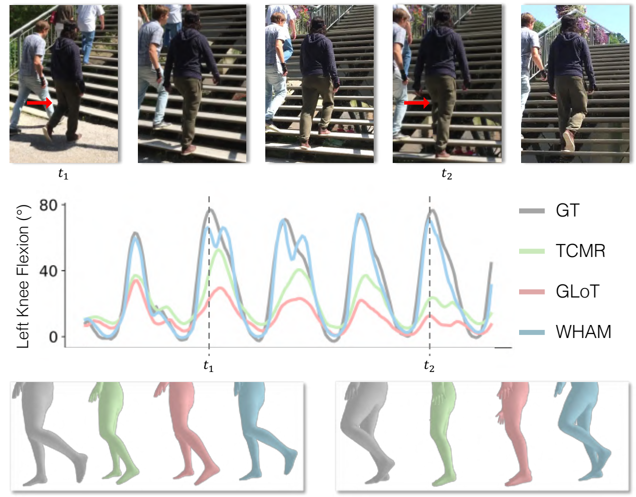

Figure 4. Qualitative comparison with previous state-of-the-art methods for 3D human pose and shape estimation. See text.

[1] Previous work follows [15] in reporting Accel in $mm/\text{frame}^2$. To remove the dependency on frame rate, we convert all previously reported results to $m/s^2$.

| | Models | 3DPW (14) | | | | RICH (24) | | | | EMDB (24) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | PA-MPJPE | MPJPE | PVE | Accel | PA-MPJPE | MPJPE | PVE | Accel | PA-MPJPE | MPJPE | PVE | Accel |
| per-frame | SPIN [21] | 59.2 | 96.9 | 112.8 | 31.4 | 69.7 | 122.9 | 144.2 | 35.2 | 87.1 | 140.3 | 174.9 | 41.3 |
| | PARE* [18] | 46.5 | 74.5 | 88.6 | – | 60.7 | 109.2 | 123.5 | – | 72.2 | 113.9 | 133.2 | – |
| | CLIFF* [25] | 43.0 | 69.0 | 81.2 | 22.5 | 56.6 | 102.6 | 115.0 | 22.4 | 68.1 | 103.3 | 128.0 | 24.5 |
| | HybrIK* [23] | 41.8 | 71.6 | 82.3 | – | 56.4 | 96.8 | 110.4 | – | 65.6 | 103.0 | 122.2 | – |
| | HMR2.0 [7] | 44.4 | 69.8 | 82.2 | 18.1 | 48.1 | 96.0 | 110.9 | 18.8 | 60.6 | 98.0 | 120.3 | 19.8 |
| | BEDLAM-CLIFF* [1] | 43.0 | 66.9 | 78.5 | 31.0 | 50.2 | 84.4 | 95.6 | 29.3 | 60.6 | 98.0 | 111.6 | 36.1 |
| temporal | PACE [20] | – | – | – | – | 49.3 | – | – | 8.8 | – | – | – | – |
| | TCMR* [6] | 52.7 | 86.5 | 101.4 | 6.0 | 65.6 | 119.1 | 137.7 | **5.0** | 79.6 | 127.6 | 147.9 | **5.3** |
| | VIBE* [17] | 51.9 | 82.9 | 98.4 | 18.5 | 68.4 | 120.5 | 140.2 | 21.8 | 81.4 | 125.9 | 146.8 | 26.6 |
| | MPS-Net* [51] | 52.1 | 84.3 | 99.0 | 6.5 | 67.1 | 118.2 | 136.7 | 5.8 | 81.3 | 123.1 | 138.4 | 6.2 |
| | GLoT* [43] | 50.6 | 80.7 | 96.4 | **6.0** | 65.6 | 114.3 | 132.7 | 5.2 | 78.8 | 119.7 | 138.4 | 5.4 |
| | GLAMR [57] | 51.1 | – | – | 8.0 | 79.9 | – | – | 107.7 | 73.5 | 113.6 | 133.4 | 32.9 |
| | TRACE* [45] | 50.9 | 79.1 | 95.4 | 28.6 | – | – | – | – | 70.9 | 109.9 | 127.4 | 25.5 |
| | SLAHMR [54] | 55.9 | – | – | – | 52.5 | – | – | 9.4 | 69.5 | 93.5 | 110.7 | 7.1 |
| | **WHAM (Res)*** | 41.7 | 65.7 | 78.7 | 6.6 | 53.1 | 91.4 | 105.6 | 5.3 | 58.9 | 90.3 | 106.1 | 5.7 |
| | **WHAM (HR)*** | 40.9 | 64.5 | 77.7 | 6.8 | 50.7 | 88.0 | 100.8 | 5.7 | 58.4 | 89.5 | 107.7 | 6.5 |
| | **WHAM (ViT)*** | 37.8 | 60.8 | 72.5 | 6.8 | 46.2 | 84.1 | 95.5 | 5.5 | 53.4 | 87.3 | 102.9 | 5.7 |
| | **WHAM-B (ViT)*** | **37.2** | **59.4** | **71.0** | 6.9 | **44.7** | **82.6** | **93.2** | 5.6 | **48.8** | **80.7** | **93.7** | 5.9 |

Table 1. Quantitative comparison of state-of-the-art models on the 3DPW [49], RICH [10], and EMDB [16] datasets. Ordering of per-frame and temporal methods is done separately by descending MPJPE on EMDB. For testing on EMDB, we follow the protocol of *EMDB 1* [16]. Parenthesis denotes the number of body joints used to compute MPJPE and PA-MPJPE, and * denotes models trained with the 3DPW training set. Bold numbers denote the most accurate method in each column. Accel is in $m/s^2$, all other errors are in $mm$.

| Models | EMDB 2 | | | | | |
|---|---|---|---|---|---|---|
| | PA–MPJPE | W-MPJPE$_{100}$ | WA-MPJPE$_{100}$ | RTE | ROE | ERVE |
| DPVO (+ HMR2.0) [7, 47] | 49.6 | 2320.9 | 662.9 | 17.5 | 44.4 | 112.8 |
| GLAMR [57] | 56.0 | 756.1 | 286.2 | 16.7 | 74.9 | 18.0 |
| TRACE [45] | 58.0 | 2244.9 | 544.1 | 18.9 | 72.7 | 370.7 |
| SLAHMR [54] | 61.5 | 807.4 | 336.9 | 13.8 | 67.9 | 19.7 |
| **WHAM (w/ DPVO [47])** | **41.9** | 446.6 | 169.0 | 8.8 | 40.4 | 14.8 |
| **WHAM (w/ DROID [46])** | **41.9** | 439.2 | 166.1 | 8.4 | 36.0 | **14.7** |
| **WHAM (w/ GT gyro)** | **41.9** | **436.4** | **165.9** | **7.1** | **26.3** | 14.8 |

Table 2. Global motion estimation accuracy on EMDB [16].

While producing smooth results, TCMR and GLoT fail to capture the bending of the left knee when the subject is ascending the stairs, while WHAM more accurately reconstructs the 3D human pose. For more qualitative results please see the Supplemental Video.

### 4.2. 3D Global Trajectory Recovery

To evaluate global trajectory recovery, we compare WHAM with the state-of-the-art methods and a baseline that combines a SLAM method (DPVO [47]) and a per-frame method (HMR2.0 [7]); see Table 2. WHAM is agnostic to the source of the camera angular velocity and we compare results using DPVO, DROID-SLAM [46] and the ground truth angular velocity (gyro).

As shown in Table 2, WHAM outperforms the existing methods on all metrics. Specifically, combining WHAM

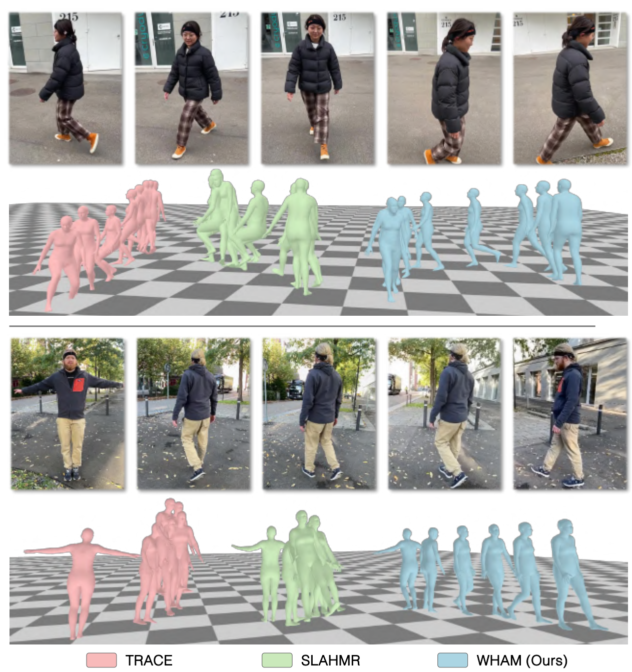


Figure 5. Qualitative comparison with TRACE [45] and SLAHMR [54] on global human motion estimation with dynamic cameras.

with DPVO is more accurate than the global trajectory esti-

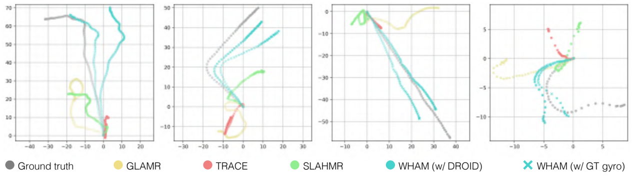


Figure 6. Comparison of global trajectory estimation on EMDB [16]. Overall, WHAM shows better alignment to ground truth data compared to GLAMR [57], TRACE [45], and SLAHMR [54].

| | EMDB 2 | | | | | |
|---|---|---|---|---|---|---|
| Methods | PA-MPJPE | PVE | RTE | ROE | ERVE | EFVE |
| w/o $F_I$ | 44.1 | 84.9 | 9.4 | 41.6 | 15.1 | 20.7 |
| w/o lifting | 108.9 | 187.5 | 18.5 | 98.3 | 29.4 | 36.8 |
| w/o traj. ref. | 41.9 | 74.7 | 9.0 | 40.4 | 14.4 | 20.9 |
| w/o $\omega$ | 40.7 | 74.5 | 18.3 | 77.1 | 14.4 | 20.8 |
| WHAM (Ours) | 41.9 | 74.7 | 8.8 | 40.4 | 14.8 | 19.8 |

Table 3. Ablation experiments. See text.

mation of DPVO combined with HMR2.0, illustrating that our method actively refines the global trajectory instead of performing a simple integration. DROID-SLAM gives slightly better results than DPVO. Furthermore, WHAM significantly outperforms TRACE on egocentric root and foot velocities. We further demonstrate this in Figs. 5 and 1, where WHAM captures more consistent and plausible human motion in the global coordinate system than TRACE and SLAHMR for videos captured by dynamic cameras. As depicted in Fig. 6, WHAM outperforms GLAMR, TRACE, and SLAHMR in capturing the pattern of human motion in the global coordinate system. See the Supplemental Video for more examples.

### 4.3. Ablation Study

To provide further insight into our approach, we conduct ablation studies to analyze the contribution of each component to the performance. As shown in Table 3, our entire system (WHAM) outperforms the different variants of WHAM that ablate a single component. To be specific, first, we observe that adding feature integration improves both motion and global trajectory estimation accuracy when compared with an ablated version without feature integration (w/o $F_I$). Similarly, the removal of the pre-training on the 2D-to-3D lifting task using AMASS [32] (w/o lifting) shows significant performance degradation. In addition, we observe that WHAM without the trajectory refinement (w/o traj. ref.) gives higher root translation and foot velocity error, indicating that our refinement approach contributes to improving the global trajectory estimation and helps reduce foot sliding. Last, we experiment with WHAM to decode trajectory solely based on the motion context without using the estimated camera angular velocity (w/o $\omega$). Although this version shows robust performance on predicting 3D human pose, it suffers from the entanglement of camera and human motion, resulting in significantly high global trajectory errors (RTE and ROE).

## 5. Conclusion

WHAM is a new method to recover accurate 3D human motion in global coordinates from a moving camera more efficiently and accurately than the state-of-the-art approaches. Our approach leverages the AMASS dataset to train a network to recursively lift 2D sequences of keypoints to sequences of 3D SMPL parameters. But keypoints alone lack valuable information about the body and its movement. Consequently, we integrate image context information over time and learn to combine it with the motion context to better estimate human body shape and pose. Additionally, our method takes an estimate of the camera angular velocity, which can either be computed from a SLAM method or from the camera's gyro when available. Finally, we combine all this information with an estimate of foot contact to recover the 3D human motion in global coordinates from a monocular video sequence. WHAM significantly outperforms the existing state-of-the-art methods (both image-based and video-based) on challenging in-the-wild benchmarks in both 3D motion and the world-coordinate trajectory estimation accuracy. Because of its speed and accuracy, WHAM provides a foundation for in-the-wild motion capture applications.

**Limitations:** WHAM learns about human motion from AMASS, limiting generalization to motions that are out of distribution. For example, since AMASS does not contain people riding bicycles or skateboards, WHAM does not capture global motion in these cases. Our contact estimation only applies to the feet and should be extended to include other body parts that may be in contact with the scene. Since WHAM relies on an estimate of the camera's angular velocity, errors in this estimate can accumulate over time, leading to drift in the global trajectory. While we employ random masking as part of our data synthesis process, our generating approach mainly assumes the scenario where the full body is within the field of view. This can be addressed with additional augmentation during training (cf. [18, 26]). See the Supplemental Video. for example failure cases.

**Future directions:** WHAM opens up many directions for future work. For example, while we use SLAM to estimate the camera's angular velocity, SLAM could also provide camera intrinsics and extrinsics as well as information about the 3D scene that could be used to enforce consistency between the scene and the human. While WHAM is an online method, designed for real-time applications, it could also initialize an optimization-based post-processing akin to bundle adjustment, which would optimize the camera, scene, and human motion together. Furthermore, a real-time and phone-based implementation of WHAM should be feasible.

## 6. Acknowledgement

We would like to sincerely appreciate Hongwei Yi and Silvia Zuffi for the discussion and proofreading. Part of this work was done when the first author was an intern at the Max Planck Institute for Intelligence System. This work was partially supported through an NSF CAREER Award (CBET 2145473) and a Chan-Zuckerberg Essential Open Source Software for Science Award.

**Disclosure**. MJB has received research gift funds from Adobe, Intel, Nvidia, Meta/Facebook, and Amazon. MJB has financial interests in Amazon, Datagen Technologies, and Meshcapade GmbH. While MJB is a consultant for Meshcapade, his research in this project was performed solely at, and funded solely by, the Max Planck Society.

## References

[1] Michael J. Black, Priyanka Patel, Joachim Tesch, and Jinlong Yang. BEDLAM: A synthetic dataset of bodies exhibiting detailed lifelike animated motion. In *Proceedings IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 8726–8737, 2023. 4, 5, 6, 7, 13

[2] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J. Black. Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In *Computer Vision – ECCV 2016*, pages 561–578. Springer International Publishing, 2016. 3

[3] Zhongang Cai, Wanqi Yin, Ailing Zeng, Chen Wei, Qingping Sun, Yanjun Wang, Hui En Pang, Haiyi Mei, Mingyuan Zhang, Lei Zhang, Chen Change Loy, Lei Yang, and Ziwei Liu. SMPLer-X: Scaling up expressive human pose and shape estimation. In *Proc. Neural Information Processing Systems (NeurIPS)*, 2023. 6

[4] 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. 12

[5] Hongsuk Choi, Gyeongsik Moon, and Kyoung Mu Lee. Pose2mesh: Graph convolutional network for 3d human pose and mesh recovery from a 2d human pose. In *European Conference on Computer Vision (ECCV)*, 2020. 3

[6] Hongsuk Choi, Gyeongsik Moon, Ju Yong Chang, and Kyoung Mu Lee. Beyond static features for temporally consistent 3d human pose and shape from a video. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2, 3, 4, 6, 7, 12

[7] Shubham Goel, Georgios Pavlakos, Jathushan Rajasegaran, Angjoo Kanazawa, and Jitendra Malik. Humans in 4D: Reconstructing and tracking humans with transformers. *arXiv preprint arXiv:2305.20091*, 2023. 2, 3, 4, 6, 7

[8] Riza Alp Guler and Iasonas Kokkinos. Holopose: Holistic 3d human reconstruction in-the-wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 3

[9] Dorian F. Henning, Christopher Choi, Simon Schaefer, and Stefan Leutenegger. Bodyslam++: Fast and tightly-coupled visual-inertial camera and human motion tracking, 2023. 3

[10] Chun-Hao P. Huang, Hongwei Yi, Markus Höschle, Matvey Safroshkin, Tsvetelina Alexiadis, Senya Polikovsky, Daniel Scharstein, and Michael J. Black. Capturing and inferring dense full-body human-scene contact. In *Proceedings IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 13274–13285, 2022. 2, 6, 7, 13

[11] 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, 2014. 2, 5, 12, 13

[12] Hanbyul Joo, Natalia Neverova, and Andrea Vedaldi. Exemplar fine-tuning for 3d human pose fitting towards in-the-wild 3d human pose estimation. In *3DV*, 2020. 3

[13] Hanbyul Joo, Natalia Neverova, and Andrea Vedaldi. Exemplar fine-tuning for 3d human pose fitting towards in-the-wild 3d human pose estimation. In *3DV*, 2020.

[14] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Computer Vision and Pattern Regognition (CVPR)*, 2018. 2, 3, 6

[15] Angjoo Kanazawa, Jason Y. Zhang, Panna Felsen, and Jitendra Malik. Learning 3d human dynamics from video. *arXiv preprint arXiv:1812.01601*, 2018. 2, 3, 5, 6, 12, 13

[16] Manuel Kaufmann, Jie Song, Chen Guo, Kaiyue Shen, Tianjian Jiang, Chengcheng Tang, Juan José Zárate, and Otmar Hilliges. EMDB: The Electromagnetic Database of Global 3D Human Pose and Shape in the Wild. In *International Conference on Computer Vision (ICCV)*, 2023. 2, 3, 6, 7, 8, 13

[17] Muhammed Kocabas, Nikos Athanasiou, and Michael J. Black. VIBE: Video inference for human body pose and shape estimation. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 5252–5262. IEEE, 2020. 2, 3, 6, 7, 12

[18] Muhammed Kocabas, Chun-Hao P. Huang, Otmar Hilliges, and Michael J. Black. PARE: Part attention regressor for 3D human body estimation. In *Proc. International Conference on Computer Vision (ICCV)*, pages 11127–11137, 2021. 2, 3, 7, 8

[19] Muhammed Kocabas, Chun-Hao P. Huang, Joachim Tesch, Lea Müller, Otmar Hilliges, and Michael J. Black. SPEC: Seeing people in the wild with an estimated camera. In *Proc. International Conference on Computer Vision (ICCV)*, pages 11015–11025, Piscataway, NJ, 2021. IEEE. 3

[20] Muhammed Kocabas, Ye Yuan, Pavlo Molchanov, Yunrong Guo, Michael J. Black, Otmar Hilliges, Jan Kautz, and Umar Iqbal. Pace: Human and motion estimation from in-the-wild videos. In *3DV*, 2024. 2, 3, 6, 7

[21] Nikos Kolotouros, Georgios Pavlakos, Michael J Black, and Kostas Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *ICCV*, 2019. 2, 3, 4, 6, 7

[22] Nikos Kolotouros, Georgios Pavlakos, and Kostas Daniilidis. Convolutional mesh regression for single-image human shape reconstruction. In *CVPR*, 2019. 3

[23] Jiefeng Li, Chao Xu, Zhicun Chen, Siyuan Bian, Lixin Yang, and Cewu Lu. 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*, pages 3383–3393, 2021. 2, 3, 6, 7

[24] Jiefeng Li, Siyuan Bian, Qi Liu, Jiasheng Tang, Fan Wang, and Cewu Lu. NIKI: Neural inverse kinematics with invertible neural networks for 3d human pose and shape estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 3

[25] Zhihao Li, Jianzhuang Liu, Zhensong Zhang, Songcen Xu, and Youliang Yan. Cliff: Carrying location information in full frames into human pose and shape estimation. In *ECCV*, 2022. 2, 3, 4, 6, 7

[26] Jing Lin, Ailing Zeng, Haoqian Wang, Lei Zhang, and Yu Li. One-stage 3d whole-body mesh recovery with component aware transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21159–21168, 2023. 8

[27] Kevin Lin, Chung-Ching Lin, Lin Liang, Zicheng Liu, and Lijuan Wang. Mpt: Mesh pre-training with transformers for human pose and mesh reconstruction, 2023. 3

[28] Miao Liu, Dexin Yang, Yan Zhang, Zhaopeng Cui, James M Rehg, and Siyu Tang. 4d human body capture from egocentric video via 3d scene grounding. In *3DV*, 2021. 2

[29] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, 2015. 3

[30] Zhengyi Luo, S. Alireza Golestaneh, and Kris M. Kitani. 3d human motion estimation via motion compression and refinement. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, 2020. 2, 3, 4, 6, 12

[31] Xiaoxuan Ma, Jiajun Su, Chunyu Wang, Wentao Zhu, and Yizhou Wang. 3d human mesh estimation from virtual markers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 534–543, 2023. 3

[32] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. AMASS: Archive of motion capture as surface shapes. In *International Conference on Computer Vision*, pages 5442–5451, 2019. 2, 3, 5, 8, 12

[33] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *3D Vision (3DV), 2017 Fifth International Conference on*. IEEE, 2017. 2, 5, 12, 13

[34] Gyeongsik Moon and Kyoung Mu Lee. I2l-meshnet: Image-to-lixel prediction network for accurate 3d human pose and mesh estimation from a single rgb image. In *European Conference on Computer Vision (ECCV)*, 2020. 3

[35] Gyeongsik Moon, Hongsuk Choi, and Kyoung Mu Lee. Neuralannot: Neural annotator for 3d human mesh training sets. In *Computer Vision and Pattern Recognition Workshop (CVPRW)*, 2022. 3, 13

[36] Gyeongsik Moon, Hongsuk Choi, Sanghyuk Chun, Jiyoung Lee, and Sangdoo Yun. Three recipes for better 3d pseudo-gts of 3d human mesh estimation in the wild. In *Computer Vision and Pattern Recognition Workshop (CVPRW)*, 2023. 3

[37] Ahmed A. A. Osman, Timo Bolkart, and Michael J. Black. STAR: Sparse trained articulated human body regressor. In *Computer Vision - ECCV 2020*, pages 598–613. Springer, 2020. 3

[38] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3D hands, face, and body from a single image. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 10975–10985, 2019. 3, 13

[39] Georgios Pavlakos, Luyang Zhu, Xiaowei Zhou, and Kostas Daniilidis. Learning to estimate 3d human pose and shape from a single color image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 459–468, 2019. 3

[40] Davis Rempe, Tolga Birdal, Aaron Hertzmann, Jimei Yang, Srinath Sridhar, and Leonidas J. Guibas. Humor: 3d human motion model for robust pose estimation. In *International Conference on Computer Vision (ICCV)*, 2021. 3

[41] Nitin Saini, Chun-Hao P. Huang, Michael J. Black, and Aamir Ahmad. Smartmocap: Joint estimation of human and camera motion using uncalibrated rgb cameras. *IEEE Robotics and Automation Letters*, pages 1–8, 2023. 2

[42] Simon Schaefer, Dorian F. Henning, and Stefan Leutenegger. Glopro: Globally-consistent uncertainty-aware 3d human pose estimation & tracking in the wild, 2023. 2, 3

[43] Xiaolong Shen, Zongxin Yang, Xiaohan Wang, Jianxin Ma, Chang Zhou, and Yi Yang. Global-to-local modeling for video-based 3d human pose and shape estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8887–8896, 2023. 2, 3, 4, 6, 7, 12

[44] Soyong Shin, Zhixiong Li, and Eni Halilaj. Markerless motion tracking with noisy video and imu data. *IEEE Transactions on Biomedical Engineering*, 70(11):3082–3092, 2023. 12

[45] Yu Sun, Qian Bao, Wu Liu, Tao Mei, and Michael J. Black. TRACE: 5D Temporal Regression of Avatars with Dynamic Cameras in 3D Environments. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1, 2, 3, 7, 8

[46] Zachary Teed and Jia Deng. DROID-SLAM: Deep Visual SLAM for Monocular, Stereo, and RGB-D Cameras. *Advances in neural information processing systems*, 2021. 3, 5, 7

[47] Zachary Teed, Lahav Lipson, and Jia Deng. Deep patch visual odometry. *arXiv preprint arXiv:2208.04726*, 2022. 3, 5, 7

[48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017. 3

[49] Timo von Marcard, Roberto Henschel, Michael Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *European Conference on Computer Vision (ECCV)*, 2018. 2, 3, 5, 6, 7, 12, 13

[50] Ziniu Wan, Zhengjia Li, Maoqing Tian, Jianbo Liu, Shuai Yi, and Hongsheng Li. Encoder-decoder with multi-level attention for 3d human shape and pose estimation. In *The IEEE International Conference on Computer Vision (ICCV)*, 2021. 3

[51] Wen-Li Wei, Jen-Chun Lin, Tyng-Luh Liu, and Hong-Yuan Mark Liao. Capturing humans in motion: Temporal-attentive 3d human pose and shape estimation from monocular video. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2, 3, 4, 6, 7, 12

[52] Hongyi Xu, Eduard Gabriel Bazavan, Andrei Zanfir, William T. Freeman, Rahul Sukthankar, and Cristian Sminchisescu. GHUM & GHUML: generative 3D human shape and articulated pose models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6183–6192, 2020. 3

[53] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. ViTPose: Simple vision transformer baselines for human pose estimation. In *Advances in Neural Information Processing Systems*, 2022. 2, 4

[54] Vickie Ye, Georgios Pavlakos, Jitendra Malik, and Angjoo Kanazawa. Decoupling human and camera motion from videos in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1, 2, 3, 6, 7, 8, 13

[55] Xinyu Yi, Yuxiao Zhou, and Feng Xu. Transpose: Real-time 3d human translation and pose estimation with six inertial sensors. *ACM Transactions on Graphics*, 40(4), 2021. 12

[56] 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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 4, 13

[57] Ye Yuan, Umar Iqbal, Pavlo Molchanov, Kris Kitani, and Jan Kautz. Glamr: Global occlusion-aware human mesh recovery with dynamic cameras. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 3, 5, 7, 8

[58] Feng Zhang, Xiatian Zhu, Hanbin Dai, Mao Ye, and Ce Zhu. Distribution-aware coordinate representation for human pose estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[59] Hongwen Zhang, Yating Tian, Yuxiang Zhang, Mengcheng Li, Liang An, Zhenan Sun, and Yebin Liu. Pymaf-x: Towards well-aligned full-body model regression from monocular images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 2, 3

[60] Yuxiang Zhang, Hongwen Zhang, Liangxiao Hu, Hongwei Yi, Shengping Zhang, and Yebin Liu. Real-time monocular full-body capture in world space via sequential proxy-to-motion learning. *arXiv preprint arXiv:2307.01200*, 2023. 2, 3, 5, 12

[61] 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. 2, 3, 4, 5

# A. Supplementary Materials

As promised in the main paper, this supplemental document provides details of our synthetic data generation, the datasets we use, our network training, and the run-time cost. Additionally, please refer to our **Supplemental Video** for results that illustrate our method and recent SOTA methods applied to video sequences.

## A.1. Synthetic data generation

To address the scarcity of the video data with paired 3D ground truth, we pretrain WHAM on an extensive number of synthetic 2D keypoint sequences for which we have the ground truth 3D poses. In this section, we describe the process of data synthesis from AMASS [32].

**2D keypoint sequence synthesis.** During training, we sample sequences of SMPL poses of length $L = 81$ from AMASS. Then, similar to MEVA [30], we uniformly upsample or downsample the frames to speed up or down the motion by up to 50% of the original speed. Furthermore, we apply a random root rotation $\Delta\Gamma \sim \mathcal{U}(0°, 360°)$ to the axis that is vertical to the ground plane and Gaussian noise to the shape parameter $\Delta\beta \sim \mathcal{N}(0, 0.1)$. Given the augmented SMPL sequence, we extract 3D keypoints that correspond to the MS-COCO keypoints and add 3D noise modeled following previous work [44]. Finally, we apply a random mask with the average probability of $p = 0.15$ to the 3D keypoints and project them onto the virtual camera as described below.

**Contact label generation.** The goal of generating a contact label is to train the motion decoder to detect the foot-ground contact accurately. Previous work [60] uses both the velocity and height of the feet to generate contact labels. However, in order to generalize our approach to arbitrary ground conditions such as slopes or stairs, we only use foot velocity to compute the ground truth contact labels, similar to TransPose [55]. We use heel and toe vertices of each foot to define foot-ground contact and compute the probability as below:

$$\hat{p}^{(t)} = \frac{1}{1 + e^{\alpha(v^{(t)} - v_t)/v_t}}.$$

We set the threshold velocity $v_t = 1cm/$frame and the coefficient $\alpha = 5$.

**Camera motion synthesis.** We begin with generating the initial pose of the virtual camera, followed by the modeling of the camera motion. We model the initial roll and pitch angles of the camera using Gaussian distributions:

$$\begin{aligned} \gamma_r^{(0)} &\sim \mathcal{N}(0°, 5°), \\ \gamma_p^{(0)} &\sim \mathcal{N}(5°, 22.5°). \end{aligned}$$

Here, we do not model the initial yaw angle since it is already handled by the random SMPL root rotation $\Delta\Gamma$.

Subsequently, we sample the initial camera translation, using a mix of unfiorm and normal distributions, to capture the ground-truth 3D pose in the camera coordinates

$$\begin{aligned} T_z^{(0)} &\sim \mathcal{U}(2m, 12m) - \left(R^{(0)}t^{(0)}\right)_z, \\ T_x^{(0)} &\sim \mathcal{N}(0, 0.25d)\ d - \left(R^{(0)}t^{(0)}\right)_x, \\ T_y^{(0)} &\sim \mathcal{N}(0, 0.25d)\ d - \left(R^{(0)}t^{(0)}\right)_y. \end{aligned}$$

Here, $d = w \cdot T_z / 2f$ is the maximum displacement of the camera to capture the 3D keypoints within the field of view, $R^{(0)}$ is the initial camera pose, $t^{(0)}$ is the initial human translation, $w$ is the image size, and $f$ is the focal length. Next, we sample the magnitude of change in the camera's extrinsics with the Gaussian distributions:

$$\begin{aligned} \Delta\gamma_y &\sim \mathcal{N}(0°, 45°), \\ \Delta\gamma_r, \Delta\gamma_p &\sim \mathcal{N}(0°, 22.5°), \\ \Delta T_x, \Delta T_y, \Delta T_z &\sim \mathcal{N}(0m, 1m). \end{aligned}$$

Finally, we interpolate the extrinsics and construct the camera's dynamic path. Here, we sample the time stamp with 20% of noise, instead of uniform sampling, to model the non-linear camera motion.

We use 6.7M frames in total and uniformly sample them during the synthetic data generation.

## A.2. Datasets

In this section, we illustrate the datasets we use for training and testing our method.

**Human3.6M** [11] is an indoor dataset containing individuals performing 15 distinct actions captured by both a motion-capture (mocap) system and 4 calibrated video cameras. The ground truth 3D keypoint locations are provided by the mocap. Following previous work [6, 17, 43, 51], we use 5 subjects (S1, S5, S6, S7, and S8) to train our network after downsampling the mocap data to 25 fps.

**MPI-INF-3DHP** [33] is a multi-view and markerless dataset containing individuals performing various ranges of motion with corresponding ground-truth 3D keypoint locations. To train the network, we use the training set of the dataset, containing 8 subjects and 16 videos per subject.

**InstaVariety** [15] is a large-scale in-the-wild video dataset with large variations in subjects, motion, and environment. The dataset contains pseudo-ground-truth 2D keypoints detected by OpenPose [4]. We train our method on the training split of the dataset.

**3DPW** [49] is an in-the-wild video dataset containing ground truth 3D pose captured by a hand-held camera and 13 body-worn inertial sensors. We use the train, validation, and test splits of 3DPW for training, validating, and testing our method.

| Methods | 3DPW PA-MPJPE | MPJPE | PVE | Accel |
|---|---|---|---|---|
| w/o Neural State Initialization | **37.8** | 64.5 | 76.3 | 6.9 |
| **WHAM** | **37.8** | **60.8** | **72.5** | **6.8** |

Table 4. Quantitative analysis of neural state initialization on the 3DPW [49] dataset.

**RICH** [10] is a large-scale multi-view dataset captured in both indoor and outdoor environments. RICH provides the ground truth SMPL-X [38] parameters. Following previous work [1], we use the test split of the dataset to evaluate our method on 3D pose estimation accuracy.
**EMDB** [16] is a recently captured dataset that uses a dynamic camera and body-worn electromagnetic (EM) sensors. EMDB provides ground-truth SMPL parameters as well as the global trajectory of the individuals in a global coordinate system. We use two distinct test splits, *EMDB 1* and *EMDB 2*, to evaluate the performance on 3D pose and shape estimation (*EMDB 1*) and global trajectory estimation (*EMDB 2*).
**BEDLAM** [1] is a recently proposed large-scale synthetic dataset. BEDLAM introduces realistic modeling of diverse clothing, hair, motion, skin tones, and scene environments to synthesize videos. BEDLAM contains 1 million video frames for individuals with ground truth SMPL/SMPL-X parameters. We optionally use the train split of BEDLAM to train the network.

### A.3. Neural-network Initialization

Uni-directional RNNs introduce the challenge of differing learning objectives between the initial frames and subsequent ones due to the initialization state. Specifically, in traditional RNNs, the initialization state is typically padded with zeros, resulting in the first frame primarily relying on the input signal. In contrast, the subsequent frames are trained to capitalize on both the input signal and information transferred from the past. To resolve the disparity in learning objectives, we use a neural initialization network, as proposed by [56], to predict $h_{0,E}$ and $h_{0,D}$ from the 0-th frame pose, instead of using zero-padding. During the training, we use pseudo-ground-truth 3D pose [35] for the video datasets that do not have the SMPL parameter annotation [11, 15, 33]. Note that we do not supervise our network on the pseudo labels. At test time, we use the pose and shape predicted by a single-frame regressor as the initial state. As shown in Table 4, we observe that the use of neural state initialization increases the performance in 3D human pose and shape estimation.

| Methods | Runtime: fps ($ms$) batch size = 1 | batch size = 64 |
|---|---|---|
| Bounding box detection | 70 (14.3 $ms$) | 265 (3.8 $ms$) |
| Bounding box tracking | 7189 (0.1 $ms$) | 7189 (0.1 $ms$) |
| 2D keypoints detection | 12.1 (82.6 $ms$) | 88 (11.4 $ms$) |
| Image feature extraction | 66 (15.2 $ms$) | 237 (4.3 $ms$) |
| Rest of the framework | 926 (1.1 $ms$) | 1431 (0.7 $ms$) |
| Total | 8.8 (113.3 $ms$) | 49.3 (20.3 $ms$) |

Table 5. Per-frame computation time (running time) of each module in WHAM. We present this both as frames per second (fps) and milliseconds ($ms$).

### A.4. Run-time cost

While the core WHAM network presented here runs at 200fps, it relies on the input of several other methods. Here we compute the run time of each module required by our framework on the EMDB dataset [16]. The inference speed of all methods was computed on a single A100 GPU. We exclude running SLAM in this analysis as it can be obviated if we use gyroscope data (though real-time SLAM methods exist). As shown in Table 5, our full method, with pre-processing steps, runs at around 9 fps with online inference (i.e., a batch size of 1 and no lag), and around 50 fps when run in batch mode (with resulting lag). We compare the core runtime of WHAM with SLAHMR [54], excluding bounding box detection, person identification, and keypoints detection, for which there are real-time solutions. In this condition, WHAM takes 5 seconds (202 fps) for 1000 frames. Specifically, WHAM takes 4.3 seconds (237 fps) for image feature extraction and 0.7 seconds (1431 fps) to regress the motion and global trajectory. This is significantly faster than SLAHMR which takes 260 minutes ($<$ 0.1 fps) per 1000 frames.