EM-POSE: 3D Human Pose Estimation from Sparse Electromagnetic Trackers

Manuel Kaufmann[1,2] Yi Zhao[2] Chengcheng Tang[2] Lingling Tao[2]
Christopher Twigg[2] Jie Song[1] Robert Wang[2] Otmar Hilliges[1]

[1]ETH Zürich, Department of Computer Science [2]Facebook Reality Labs

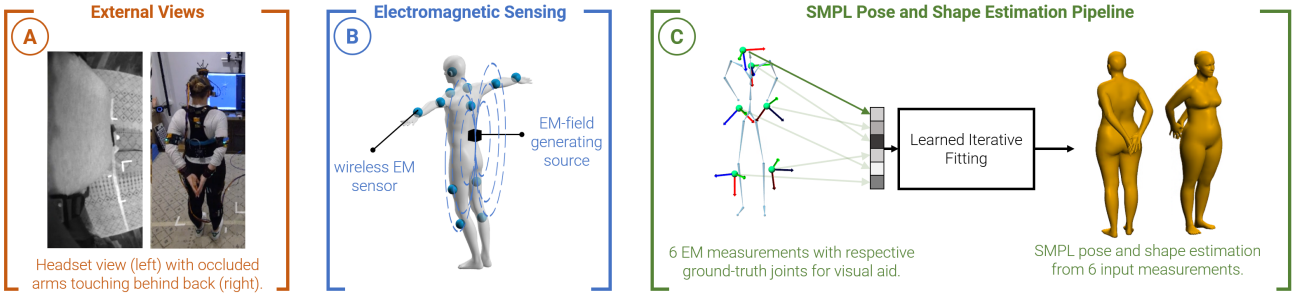

Figure 1: Reconstructing the subject's full-body pose is important to create immersive experiences in AR/VR. While external cameras limit the capture space and head-worn cameras can suffer from heavy self-occlusions in top-down views (A), our method reconstructs the body pose from electromagnetic (EM) field-based sensing (B). We leverage a customized system consisting of up to 12 wireless sensors measuring their 6D pose relative to a body-worn source. We adopt learned gradient descent (LGD) [53] to estimate SMPL pose and shape from as little as 6 EM sensors (C) tested on a newly captured dataset.

Abstract

Fully immersive experiences in AR/VR depend on reconstructing the full body pose of the user without restricting their motion. In this paper we study the use of body-worn electromagnetic (EM) field-based sensing for the task of 3D human pose reconstruction. To this end, we present a method to estimate SMPL parameters from 6-12 EM sensors. We leverage a customized wearable system consisting of wireless EM sensors measuring time-synchronized 6D poses at 120 Hz. To provide accurate poses even with little user instrumentation, we adopt a recently proposed hybrid framework, learned gradient descent (LGD), to iteratively estimate SMPL pose and shape from our input measurements. This allows us to harness powerful pose priors to cope with the idiosyncrasies of the input data and achieve accurate pose estimates. The proposed method uses AMASS to synthesize virtual EM-sensor data and we show that it generalizes well to a newly captured real dataset consisting of a total of 37 minutes of motion from 5 subjects. We achieve reconstruction errors as low as 31.8 mm and 13.3 degrees, outperforming both pure learning- and pure optimization-based methods. Code and data is available under https://ait.ethz.ch/projects/2021/em-pose.

1. Introduction

AR and VR (collectively called XR) is a promising new computing platform for entertainment, communication, medicine, remote presence and more. An important component of an immersive XR system is a method to accurately reconstruct the full body pose of the user. While external camera-based pose estimation has progressed at a rapid pace (*e.g.*, [14, 19, 21, 59]) such approaches inherently limit the mobility of the user due to the requirement for external cameras. Body-worn tracking using inertial-measurement units (IMUs) [17, 33, 45, 49, 64, 65] or cameras [48, 51, 57, 69] allow for free movement, but suffer from lack of accurate positional measurements in the case of IMUs, and heavy occlusions for camera-based systems, resulting in incorrect pose estimates that may drift over time.

In this paper we propose a new approach to body-worn pose estimation that is based on electromagnetic-field (EM) sensing which can replace or complement vision or IMU-based counterparts. In our method an EM field is emitted from a source that is worn on the body and a small number of sensors measure their position and orientation relative to the emitted magnetic field (*c.f.* Fig. 1). In our implementation, we leverage a fully wireless magnetic tracking system consisting of up to 12 sensors. These sensors are small (roughly half the size of a credit card), low-powered, and

have been customized to enable accurate tracking of fast, dynamic motions at update rates up to 120 Hz. Compared to optical tracking, our sensors are typically within 1 cm positional and 2-3 degrees angular error.

However, reconstructing the full articulated pose from these measurements with high accuracy remains difficult du to several challenges. First, for a convenient system, only a small number of body-worn sensors should be used, making the pose estimation problem underconstrained. We show good accuracy with as little as 6 sensors. Second, the accuracy of the position and orientation measurements depend on the distance of the sensor to the source. So, under dynamic human motion, the sensor accuracy varies as a function of pose. Third, the skin-to-sensor offsets must be determined. These offsets can vary due to possible slipping of the sensor against the skin. Hence, the resulting method should be robust to changes in these offsets.

Embracing these challenges, we propose a new EM-based pose estimation method that leverages the recently proposed learned gradient descent (LGD) [53] framework to iteratively fit a parametric body model, here SMPL [30], to the EM measurements, where the parameter update rule is predicted by a neural network. The method is based on the key insights that the sensor measurements are perturbed by dynamically varying sources of noise: EM-interference, pose dependent effects, and offsets to the underlying joints. The parametric body model in combination with a learned parameter update rule allows us to integrate strong priors into the pose estimation pipeline. Furthermore, with LGD the parameter updates stay on the manifold of valid poses thus allowing for larger step sizes leading to fast convergence in few steps. SMPL enables us to synthesize virtual positions and orientations on the skin, which we leverage to train LGD on AMASS [32] by simulating many pairs of virtual EM sensors and SMPL references. To close the gap between synthetic and real data, we extract estimates of subject-specific skin-to-sensor offsets from a designated calibration sequence. These offsets are used during training to adjust and augment the synthetic data. Our evaluations show that the proposed method generalizes well to a newly recorded dataset without requiring fine-tuning, even for subjects whose offsets were not seen during training.

To foster future research into this direction, we release a new dataset containing pairs of magnetic measurements and SMPL poses. We obtained SMPL reference poses via multi-view tracking from outside-in RGB-D data together with manual annotations. The dataset consists of 45 sequences of a total length of 37.1 minutes and was recorded with 3 female and 2 male participants. In our evaluations we achieve average reconstruction errors of 31.8 mm and 13.3 ° with 12 sensors and 35.4 mm and 14.9 ° with 6 sensors. In comparative experiments we show that this outperforms the state-of-the-art in optimization-based approaches to register SMPL to motion-capture markers [32], a specialized optimization method for EM data and a hard learning-based baseline, inspired by IMU-based prior work [17].

We see our system as complementary to pure vision-based methods. Because it is light-weight, low-powered, wireless, and accurate, it potentially enables the collection of in-the-wild datasets - currently the biggest challenge for RGB-based methods because of a lack of data. It can also be used to collect reference poses when image data is affected by occlusions or motion blur, *e.g.* in egocentric views.

In summary, in this paper we contribute i) a method to estimate SMPL pose and shape parameters from as little as 6 EM sensors leveraging a customized wearable EM sensing based system ii) a general framework to estimate SMPL parameters from few on-skin measurements which is agnostic to the underlying sensing technology, and iii) a dataset consisting of EM sensor data and SMPL pose pairs. Code and data are available under https://ait.ethz.ch/projects/2021/em-pose.

2. Related Work

Inertial Tracking Pose estimation from inertial measurement units (IMUs) is popular because modern IMUs are small and do not require line-of-sight (LoS). They do however suffer from drift, which commercial systems like Xsens [49] mitigate by employing a high number of sensors in conjunction with biomechanical body models. Other works use body-worn acoustic sensors to provide inter-sensor distance measurements, *e.g.* [28, 63] or fuse IMUs with external camera views, *e.g.* [6, 11, 33, 44, 45, 58, 64, 71]. This works well but increases instrumentation, limits the capture space, and re-introduces LoS constraints. To ease usability researchers have also investigated reducing the required amount of sensors, *e.g.* [7, 17, 64, 65]. This however, leaves the pose heavily underconstrained necessitating either costly optimizations [65], an external camera [64] or fine-tuning a neural network on real data [17]. SIP/DIP [17] are the closest work to ours in spirit as we also leverage AMASS [32]. However, our hybrid method is considerably faster at runtime than SIP, and unlike DIP does not require fine-tuning and can handle multiple subjects all while achieving errors that are lower than what was reported by DIP. In summary, IMUs are inherently limited by the fact that they do not observe position directly and drift over time - a circumstance that magnetic systems rectify.

Optical and Related Tracking Optical tracking of spherical retro-reflective markers, *e.g.* [38, 62], yields high accuracy and update rates, but requires LoS and typically many (40+) markers. Researchers have investigated the use of physically-based models to solve for pose [75], how to clean up raw marker data [4, 9, 16, 25, 41], or using large

marker sets to capture skin deformation [39]. More recently, the availability of statistical 3D human body models, *e.g.* [1–3, 31, 46] have allowed methods such as MoSh [29] or MoSh++ [32] to fit pose and shape to sets of around 40 markers thus enabling the unification of several motion capture databases into a large-scale dataset named AMASS [32]. We also reconstruct pose and shape from measurements on the skin. However, we do so from as few as 6-12 sensors and without LoS requirements. This is not only possible because our specialized hardware measures both position and orientation, but also thanks to AMASS which we leverage as a prior where pose and shape are not observed by our reduced sensor set. Recently, works have emerged using radio frequency signals, *e.g.* [26, 66, 72, 73]. This modality can traverse heavy occlusions, but again necessitates external capture equipment.

EM Tracking Uses of EM tracking technology dates back to military applications in the 1960s [42]. Ever since, it has matured considerably [47] and has achieved 6D non-LoS tracking with millisecond latency allowing applications ranging from digital input devices [8, 23, 27, 68] to medicine [56]. Naturally, it has also been applied to full-body motion capturing. The work by Roetenberg *et al.* [50] has a similar mobile setup to ours where the magnetic source is placed on the subject's lower back. However, their system is fully tethered, only applied to a few sensors and has a low update rate of 1-2 Hz. EM-based systems are tuned to working within a given range and a certain accuracy. Various commercial systems for motion capture of full-body or hands have been developed (*e.g.*, [36, 43]), but their properties are often not ideal for motion capturing with body-worn sensors. We discuss more details and differences to our customized system in Sec. 3.

Camera-based Fueled by deep neural networks, significant advances have been made in estimating 3D human pose from one or multiple RGB images, *e.g.* [18, 34, 54, 67]. Modern approaches - which often use parametric body models - tend to fall into three groups: Direct parameter regression with neural networks [13, 20, 37, 55, 59, 61, 70, 74], optimization-based techniques [12, 15, 24, 40, 52, 60], or hybrid combinations [22, 53]. We borrow ideas from the camera-based literature and adapt LGD proposed by [53] to estimate SMPL pose and shape from sparse EM measurements. Methods using head-worn cameras [48, 57, 69] allow for more mobility of a subject compared to external cameras. However, devices can be bulky and the image data can be subject to self-occlusions. In contrast, our body-worn EM-based wireless system has a small form factor and is not impacted by occlusions.

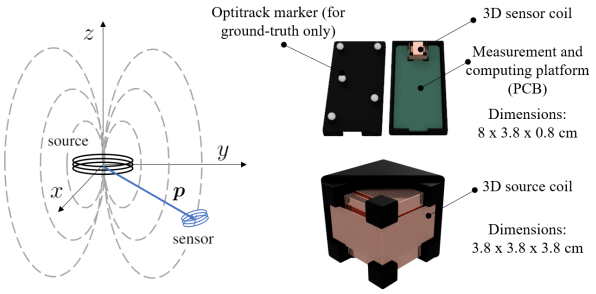

Figure 2: **EM sensing.** (left) A 1D coil is generating a magnetic B-field. Another coil can solve for its position $\boldsymbol{p}$ w.r.t. the source by comparing measured and theoretical voltage. (right) Schematics of our source and sensors.

3. Electromagnetic Sensing Hardware

Our main contribution is a method to reconstruct the full body pose from as few as 6 EM field sensors. Here and in Fig. 2 we provide a brief primer on EM sensing and summarize our hardware implementation. In Sec. 6.1 we evaluate our sensors' accuracy in a typical usage scenario.

3.1. Sensing Principle

An EM field sensing system consists of an emitter that generates magnetic fields and one or more sensors that read voltages induced by the field to estimate 6D pose. The emitter comprises of three orthogonal coils which generate three alternating current magnetic fields typically operated at kHz frequencies. The sensor, which also has three orthogonal coils, measures the voltage induced by each of the generated magnetic fields within the tracking volume. The theoretical voltages induced to each of the 3 axes of the sensor by each of the 3 emitter coils can be represented analytically via a physical model relating voltage and the pose of the sensor.

$$\boldsymbol{B}_k(\boldsymbol{p},t) = \frac{\mu_0}{4\pi}\left[\frac{3(\boldsymbol{M}_k\cdot\boldsymbol{p})\boldsymbol{p}}{|\boldsymbol{p}|^5} - \frac{|\boldsymbol{p}|^2\boldsymbol{M}_k}{|\boldsymbol{p}|^3}\right] e^{-j\omega_k t} \quad (1)$$

$$V_{k\ell}(\boldsymbol{p},\boldsymbol{R},t) = -j\omega_k n a \boldsymbol{B}_k(\boldsymbol{p},t)\cdot(\boldsymbol{R}\boldsymbol{N}_\ell) \quad (2)$$

Here $\boldsymbol{p}$ and $\boldsymbol{R}$ are the sensor position and rotation, $\boldsymbol{N}_\ell$ is the orientation of sensor axis coil ℓ, $\boldsymbol{M}_k$ is the magnetic moment of emitter axis coil k, t is time, and the remaining parameters are EM field related pre-determined parameters.

We can solve for the 6D pose $(\boldsymbol{p}(t), \boldsymbol{R}(t))$ in the least squares sense by minimizing the measured voltage $\hat{V}$ and the model voltage V along each emitter and sensor axis, *i.e.*, $\arg\min_{\boldsymbol{p}(t),\boldsymbol{R}(t)} \sum_{k=1}^{3}\sum_{\ell=1}^{3} \|\hat{V}_{k\ell}(t) - V_{k\ell}(\boldsymbol{p},\boldsymbol{R},t)\|_2^2$.

3.2. Wireless Magnetic Sensors

Magnetic tracking has been used for a variety of motion capture tasks, including hand tracking [10] and sports

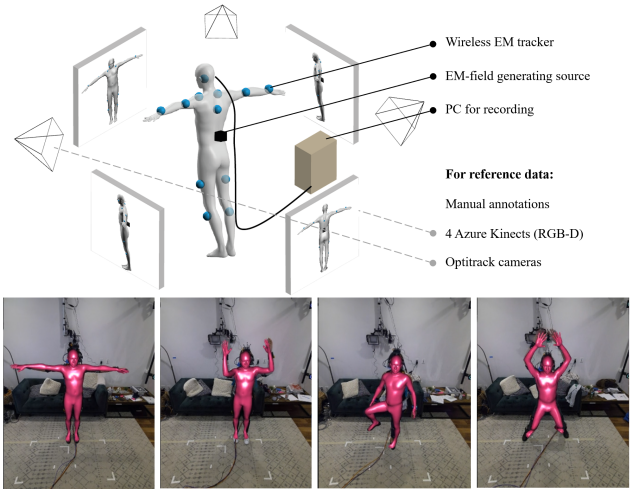

Figure 3: **Capture setup.** (Top) Overview of our capture setup to collect our real test set $\mathcal{T}$. (Bottom) Example frames of our reference data.

analytics [5]. Previous magnetic tracking systems either involve large sensors (*e.g.*, Razer Hydra) or are tethered to a PC (*e.g.*, Polhemus Liberty). Neither solution is ideal for body tracking as both large sensors and wires encumber movement. We developed a custom EM tracking system with small wireless sensors. The goal of our design is to optimize accuracy for the specified application (body tracking) within the application's constraints (small and wireless). We encountered two major challenges. The first was achieving a small form factor while retaining accurate sensing. To address this, we miniaturized the 3-axis sensing coils and carefully chose components to minimize EM interference. To achieve real-time rates with limited compute and memory, we use a piece-wise linear approximation of the voltage measurement of the EM field (*c.f.* Eq. (2)). We calibrate this function to the region of interest for our application (0.3m - 1m). The second challenge is to synchronize 12 wireless sensors and to enable communication at 120Hz with the host in real-time, while minimizing packet loss and latency. Off-the-shelf usage of the Bluetooth Low Energy (BLE) protocol is insufficient since it only supports 7 point-to-point connections and no synchronization. We designed a custom communication protocol on top of a BLE chipset that maintains microsecond synchronization among all devices with a network topology consisting of two hubs that connect to six sensors each.

4. System Overview

In this section we describe our capture setup and how it is used to obtain reference data. Please refer to Fig. 3 for an overview and the video for qualitative examples.

4.1. Capture Setup

Participants wear a customized mocap suit to attach sensors, and a customized see-through headset. We mount 12 wireless EM sensors on the body as shown in Fig. 3. Since the EM field generator is relatively small, it can be attached to the subject's lower back. All sensors except the head sensor, which is glued to the VR headset, are attached using a reusable elastic cloth band and velcro. Two communication hubs that connect wirelessly to the 12 sensors are mounted on the headset. These hubs can transmit all sensor measurements wirelessly to a nearby host. Since we simultaneously capture reference data however, we use a wired connection to a host that handles additional capture-related tasks.

To acquire reference data our capture setup uses 4 RGB-D cameras to observe the subject's motion from an outside-in viewpoint. The capture space is roughly 4 by 4 meters large and all sensing devices are time synchronized to microsecond precision. For each capture session, we calibrate the headset and RGB-D cameras, as well as the EM system so that all sensing devices share the same tracking frame, which we chose to be the Optitrack frame.

4.2. Reference Data Acquisition

In the following we give an overview of our multi-stage optimization procedure that uses 4 RGB-D cameras and the 12 EM sensors to collect reference SMPL parameters.

Body Scale We first infer body scale (*i.e.*, height and limb length) from a dedicated calibration sequence which includes a T-pose and head and limb rotations. To disambiguate the palm orientation, we manually annotate 2D hand-keypoints on a few hand-picked frames of the calibration sequence. Then we track this sequence over time and solve for body scale, using 2D-body-landmark predictions from multi-view RGB-D data, and manual hand-keypoint annotations. Once scale is established, we solve an optimization problem across multiple frames to estimate the sensor-to-body offsets to be used in the subsequent stage.

Tracking Next, we fix the body scale and sensor-to-body offsets and optimize for the body pose at each frame of the subject's sequences. Each EM sensor provides position and orientation constraints, which we augment with closest point constraints targeting the multi-view depth data. Fusing the EM tracking and depth allows us to combine the advantages of each approach: the EM sensors easily handle challenging occlusions, while the depth data helps constrain regions such as the shoulder/scapula where EM sensors are absent. We use an in-house body model, which is then converted to SMPL by [35]. We show a few illustrative examples of our reference data in Fig. 3 and the video.

Test set $\mathcal{T}$ We record a total of 45 test sequences with 5 subjects (3 female, 2 male). The recorded sequences include range-of-motion type of actions for upper and lower body, but also more natural scenarios like walking, lunges,

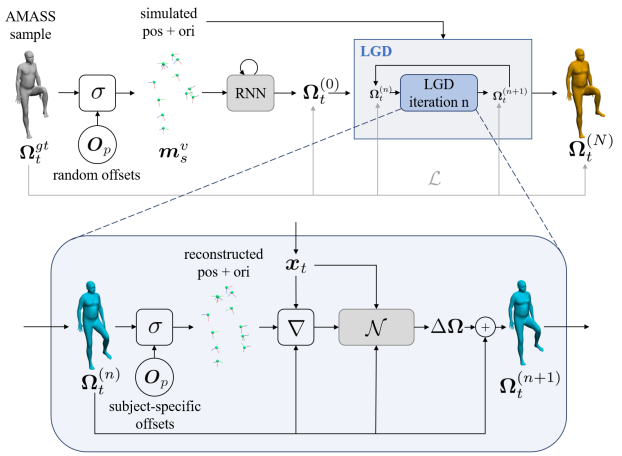

Figure 4: **Method Overview.** Given a frame from an AMASS sequence with body parameters $\mathbf{\Omega}_t^{gt}$ we randomly pick subject-specific offsets $\boldsymbol{O}_p$ to simulate S sensor positions and orientations $\boldsymbol{m}_s^v$. An RNN produces the initial estimate $\mathbf{\Omega}_t^{(0)}$, which LGD refines in N iterations to produce the final estimate $\mathbf{\Omega}_t^{(N)}$. In each iteration of LGD we compute the reconstruction loss Eq. (6) and its gradient $\nabla = \partial \mathcal{L}_r / \partial \mathbf{\Omega}_t^{(n)}$. This gradient is fed to neural network $\mathcal{N}$ and a new estimate $\mathbf{\Omega}_t^{(n+1)}$ is obtained with Eq. (5). At test time we simply feed real sensor data $\boldsymbol{m}_s$ instead of $\boldsymbol{m}_s^v$.

or jumping jacks (*c.f.* supplementary material for more details). We downsample the magnetic data from 120 Hz to 30 Hz to match the RGB-D streams. Our test set $\mathcal{T}$ thus amounts to 37.1 minutes (approx. $67,000$ frames).

5. Method

We first define our problem formally in Sec. 5.1. Then we describe in Sec. 5.2 how we synthesize virtual markers on AMASS sequences to train the LGD-based architecture shown in Sec. 5.3. Please refer to Fig. 4 for an overview.

5.1. Problem Statement

Our goal is to estimate SMPL pose and shape from sequences of EM measurements. Let the 6D pose of an EM sensor s in world space be $\boldsymbol{m}_s = (\boldsymbol{p}_s, \boldsymbol{R}_s)$. We concatenate the measurements of S sensors into a vector $\boldsymbol{x}_t = [\boldsymbol{m}_1, \dots, \boldsymbol{m}_S]$ representing a full measurement at time step t. Several measurements are summarized into a sequence $\boldsymbol{X}_i = [\boldsymbol{x}_1, \dots, \boldsymbol{x}_T]$. For each $\boldsymbol{x}_t$ we want to infer SMPL pose $\boldsymbol{\theta}_t \in \mathbb{R}^{J \cdot 3}$ and shape $\boldsymbol{\beta} \in \mathbb{R}^{10}$. With our sensor placement, we do not observe hand and foot articulation, *i.e.* $J = 19$. Although we recorded root translation, we do not consider it here, *i.e.*, we only predict global root pose.

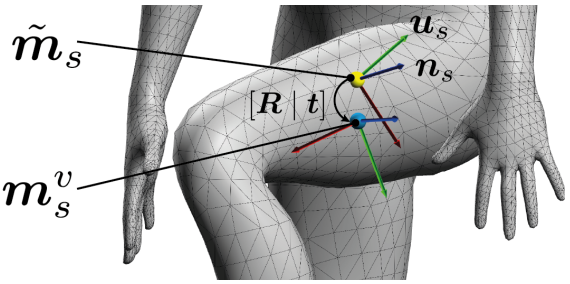

Figure 5: **Virtual sensors.** An example of a virtual position and orientation $\boldsymbol{m}_s^v$ and the offset relating it to $\tilde{\boldsymbol{m}}_s$.

5.2. Virtual Sensors

Learning the relationship between measurements $\boldsymbol{x}_t$ and pose and shape $(\boldsymbol{\theta}_t, \boldsymbol{\beta})$ would require a large-scale dataset with real EM measurements and SMPL references, which is expensive to acquire. Instead, we use AMASS [32] to synthesize virtual sensor data $\boldsymbol{x}_t^v$, described in the following.

Consider SMPL pose and shape parameters $\mathbf{\Omega} = (\boldsymbol{\theta}, \boldsymbol{\beta})$, omitting time step t for brevity. We denote the function that extracts virtual sensors as σ, *i.e.* $\boldsymbol{m}_s^v = \sigma(\mathbf{\Omega})$, where $\boldsymbol{m}_s^v = (\boldsymbol{p}_s^v, \boldsymbol{R}_s^v)$. The process is the same for all S sensors and without loss of generality we discuss a single sensor s.

In function σ we first evaluate the SMPL model on $\mathbf{\Omega}$ to obtain the corresponding mesh. For the synthesis process, we have manually pre-determined IDs of those SMPL vertices that are closest to the real mounting locations of our sensors. This only needs to be done once. To simulate $\boldsymbol{p}_s^v$ we can then simply use the vertex position $\boldsymbol{v}_s$ of the corresponding vertex ID for sensor s. Next, to simulate $\boldsymbol{R}_s^v$, we construct a local coordinate frame as follows. First, we compute the vertex normal $\boldsymbol{n}_s$ at location $\boldsymbol{v}_s$ and choose a random but fixed outgoing triangle edge $\boldsymbol{e}_s$ of unit length. We then compute $\boldsymbol{u}_s = (\boldsymbol{n}_s \times \boldsymbol{e}_s) / ||\boldsymbol{n}_s \times \boldsymbol{e}_s||_2$. Thus, we end up with the following virtual 6D pose for sensor s

$$\tilde{\boldsymbol{p}}_s = \boldsymbol{v}_s, \quad \tilde{\boldsymbol{R}}_s = \left[\frac{\boldsymbol{u}_s \times \boldsymbol{n}_s}{||\boldsymbol{u}_s \times \boldsymbol{n}_s||_2}, \boldsymbol{u}_s, \boldsymbol{n}_s \right] \tag{3}$$

which we summarize as $\tilde{\boldsymbol{m}}_s = (\tilde{\boldsymbol{p}}_s, \tilde{\boldsymbol{R}}_s)$. We could now simply equate $\boldsymbol{m}_s^v$ with $\tilde{\boldsymbol{m}}_s$ and train our method on this virtual data. If we were to do so, we would however have little chance of generalizing to real data. This is because the real sensor positions are offset by a certain amount from the skin. Furthermore, sensors are not always mounted exactly the same way and hence the hand-picked vertices $\boldsymbol{v}_s$ are only a coarse approximation. Similarly, the constructed coordinate frame $\tilde{\boldsymbol{R}}_s$ most likely does not correspond to the sensor's real orientation $\boldsymbol{R}_s$. Hence, for each sensor we model translational and rotational offsets to obtain the final virtual sensor data:

$$\boldsymbol{R}_s^v = \tilde{\boldsymbol{R}}_s \boldsymbol{R}, \quad \boldsymbol{p}_s^v = \tilde{\boldsymbol{p}}_s + \tilde{\boldsymbol{R}}_s \boldsymbol{t} \tag{4}$$

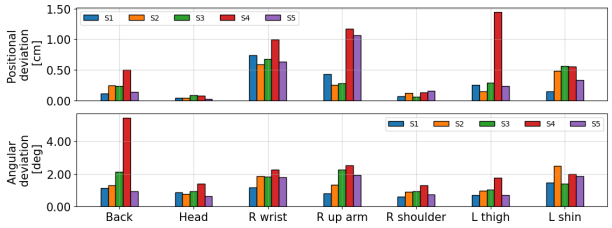

Figure 6: **Median positional and angular disagreement** between Optitrack and our EM-based system. Computed for 5 test subjects and 7 representative sensors.

For a visual depiction please refer to Fig. 5. We summarize the offsets of one sensor s as $\boldsymbol{o}_s = [\boldsymbol{R} \mid \boldsymbol{t}]$ and the collection of all S sensor offsets for a subject p as $\boldsymbol{O}_p = \{\boldsymbol{o}_s\}_{s=1}^{S}$. Note that these offsets are subject dependent, *i.e.* the full signature of $\sigma(\cdot)$ is $\boldsymbol{m}_s^v = \sigma(\boldsymbol{\Omega}, \boldsymbol{o}_s)$. Furthermore, $\boldsymbol{O}_p$ affects both pose *and* shape. Hence, any method attempting to reconstruct full-body pose and shape should choose $\boldsymbol{O}_p$ carefully. We do so by automatically extracting an estimate of $\boldsymbol{O}_p$ for each subject from a designated calibration sequence taken from $\mathcal{T}$ (*c.f.* Sec. 4.1). Please refer to the supplementary material for more details on the computation of $\boldsymbol{O}_p$. Lastly, note that these offsets are not necessarily perfectly constant over time. This is because 1) the accuracy of the magnetic sensors is range-dependent 2) sensors might move on the skin during pose articulation and 3) a hand-picked SMPL vertex $\boldsymbol{v}_s$ is not guaranteed to move in perfect synchronization with a real point on the skin.

5.3. LGD-based SMPL fitting

Using a custom variant of LGD [53] we iteratively fit SMPL parameters to our input observations $\boldsymbol{x}_t$. At training time, $\boldsymbol{x}_t$ corresponds to virtual data $\boldsymbol{x}_t^v$ whereas at test time it is the real data. LGD replaces the gradient update rule of standard gradient descent with a learned update rule which is invoked a total of N times. Assume an estimate $\boldsymbol{\Omega}_t^{(n)}$ is given. The LGD update rule at iteration n then states

$$\boldsymbol{\Omega}_t^{(n+1)} = \boldsymbol{\Omega}_t^{(n)} + \alpha \cdot \mathcal{N}(\frac{\partial \mathcal{L}_r}{\partial \boldsymbol{\Omega}_t^{(n)}}, \boldsymbol{\Omega}_t^{(n)}, \boldsymbol{x}_t) \tag{5}$$

Here $\mathcal{N}$ is a pre-trained neural network, $\alpha \in \mathbb{R}$ the step size, and $\mathcal{L}_r$ the so called reconstruction function. $\mathcal{L}_r$ measures how well our inputs can be reconstructed from the current parameter estimate $\boldsymbol{\Omega}_t^{(n)}$. It is defined as:

$$\mathcal{L}_r(\boldsymbol{x}_t, \boldsymbol{\Omega}_t^{(n)}, \boldsymbol{O}_p) = \sum_{s=1}^{S} ||\boldsymbol{m}_{t,s} - \sigma(\boldsymbol{\Omega}_t^{(n)}, \boldsymbol{o}_s)||_2^2 \tag{6}$$

where $\boldsymbol{m}_{t,s}$ are our inputs and σ computes the sensor positions and orientations given $\boldsymbol{\Omega}_t^{(n)}$ (*c.f.* Sec. 5.2).

Model	MPJPE [mm]	PA-MPJPE [mm]	MPJAE [°]
MoSh++ 12 [32]	56.9 ± 56.1	43.5 ± 33.6	21.8 ± 15.4
pos + ori 12	**44.2** ± **30.0**	**23.6** ± **13.7**	**15.4** ± **9.8**

Table 1: **Optimization-based baselines** when using all (12) input sensors. Positional and angular error on real test set.

To reap the benefits of LGD we must train the neural network $\mathcal{N}$. In contrast to [53], our input data is sequential. Hence, we first feed the inputs $\boldsymbol{x}_t$ to an RNN which produces the initial estimate $\boldsymbol{\Omega}_t^{(0)}$. This estimate is then handed over to LGD which iteratively refines it according to Eq. (5) to produce the final output $\boldsymbol{\Omega}_t^{(N)}$.

Since we want to support pose estimation for multiple subjects with a single network, we augment the virtual training data as follows: For each AMASS sequence with parameters $\boldsymbol{\Omega}_t^{gt}$ we randomly decide on a participant p whose offsets $\boldsymbol{O}_p$ should be applied. Once p is fixed, we use their offsets by feeding them to σ and thus obtain augmented virtual sensor data $\boldsymbol{x}_t^v$. At test time, we simply use the offsets corresponding to the actual subject. For training we supervise the reconstruction cost, body pose and shape at every step of the iterative refinement. In addition to [53] we also add a loss on the SMPL 3D joints $\boldsymbol{J}_t$. The loss function for time step t, iteration n and subject p is thus

$$\begin{aligned}\mathcal{L}_{n,t} =& \lambda_1 \mathcal{L}_1(\boldsymbol{\theta}_t^{(n)}, \boldsymbol{\theta}_t^{gt}) + \lambda_2 \mathcal{L}_2(\boldsymbol{\beta}^{(n)}, \boldsymbol{\beta}^{gt}) + \\ & \lambda_3 \mathcal{L}_3(\boldsymbol{J}_t^{(n)}, \boldsymbol{J}_t^{gt}) + \lambda_4 \mathcal{L}_r(\boldsymbol{x}_t, \boldsymbol{\Omega}_t^{(n)}, \boldsymbol{O}_p) \\ \mathcal{L} =& \frac{1}{NT} \sum_{n=1}^{N} \sum_{t=1}^{T} \mathcal{L}_{n,t}\end{aligned}$$

Note that to obtain a single shape estimate $\boldsymbol{\beta}^{(n)}$ per sequence we average frame-wise estimates of the shape before feeding it to the loss function. The sub-losses $\mathcal{L}_1$ to $\mathcal{L}_3$ are all the MSE. For more details on training and hyperparameters please refer to the supplementary material.

6. Evaluation

We first evaluate the accuracy of our EM-based system on a sensor level. We then compare our method to optimization- and learning-based baselines, before showing extensive ablation studies that highlight the contributions of our method. Finally, we visualize examples.

6.1. Magnetic Tracking Accuracy

To compute the accuracy of our EM-based system on a per-sensor level and in typical usage scenario we glue an Optitrack rigid body to every sensor (*c.f.* Fig. 2). Hence, for every sensor s and every time step t we obtain four measurements: its 6D pose according to Optitrack, *i.e.* $\boldsymbol{p}_s^O(t)$ and $\boldsymbol{R}_s^O(t)$, and according to the EM system, *i.e.* $\boldsymbol{p}_s^M(t)$ and

Model	MPJPE [mm]	PA-MPJPE [mm]	MPJAE [°]
ResNet 6	39.3 ± 25.4	29.6 ± 20.1	16.6 ± 11.2
BiRNN 6	36.3 ± **21.2**	27.7 ± 17.1	15.4 ± 10.2
Ours (LGD RNN) 6	**35.4** ± 21.3	**27.0** ± **16.3**	**14.9** ± **10.0**
ResNet 12	41.5 ± 27.6	30.9 ± 21.7	14.6 ± 9.8
BiRNN 12	37.3 ± 24.1	28.5 ± 18.6	14.1 ± **9.1**
Ours (LGD RNN) 12	**31.8** ± **21.0**	**24.8** ± **16.4**	**13.3** ± 9.2

Table 2: **Quantitative evaluations.** We compare our proposed hybrid method to pure learning baselines using 6 and 12 sensors. Positional and angular error on real test set.

$\boldsymbol{R}_s^M(t)$. All measurements are calibrated to world space. By design, a constant rigid transformation $[\boldsymbol{R} \mid \boldsymbol{t}]$ relates the optical and magnetic 6D pose. We can thus characterize the EM system's accuracy by computing a rigid transformation between the magnetic and optical 6D pose and measure its change over time. This boils down to solving an orthogonal Procrustes problem, the details of which are supplied in the supplementary material. This way we obtain a positional and angular error, $e_s^{pos}(t)$ and $e_s^{ang}(t)$, for every time step t. We plot the median value computed on the "jumping jacks" sequence of each subject in Fig. 6. Errors are typically around or lower than 1 cm positional and 2-3 degrees angular error. Sensors that are far away from the source (*i.e.* wrist, shin) or undergo faster motion (*i.e.* arms) experience the highest errors. In contrast, static or slow moving sensors (*i.e.* head, shoulders) show errors below 0.25 cm or 1 degree respectively. An outlier is subject 4 with sometimes high errors. This can be explained by calibration errors and degraded optical tracking when occlusions happen unexpectedly under dynamic motions, *e.g.* due to lose clothing.

6.2. Quantitative Performance

To evaluate our method quantitatively we report three common metrics: the mean per-joint positional error with and without Procrustes alignment (PA-MPJPE vs MPJPE) and the mean per-joint angular error computed on root-relative orientations (MPJAE).

Our data set $\mathcal{T}$ and method are to the best of our knowledge, the first of their kind. Therefore, no existing baseline method exists that could be applied directly to our data. The closest related work is MoSh++ [32] which estimates SMPL pose and shape from dense optical marker positions. We run our data through MoSh++ and discuss results in the following. SIP [65] and DIP [17] are more difficult to apply to our data as they require acceleration inputs which our sensors do not directly measure. Furthermore, SIP/DIP cannot estimate shape from the measurements alone. We compare to DIP approximately by adopting a similar architecture and evaluating it on $\mathcal{T}$. Furthermore we report the same metrics as DIP/SIP (PA-MPJPE, MPJAE) computed on the 15 major joints of SMPL. The results presented here are evaluated on all sequences of the first 4 of our 5 participants. We leave out subject 5 for an additional study shown in Sec. 6.4. Additionally, we also compare to an RGB-based pose estimator, VIBE [21], in the supplementary material. Finally, the EM sensors sometimes drop frames and hence we evaluate only on frames where all sensor data is available.

Optimization baselines Tab. 1 summarizes the results of two optimization baselines. To run our data through MoSh++ we supply the positional data of all 12 sensors as MoSh++ cannot take orientations into account. Not unexpectedly, the results indicate that Mosh++ struggles with this kind of data. MoSh++ was designed to produce high-quality SMPL registrations from dense optical marker arrays attached directly to the skin. Handling only 12 surface points that are neither skin-tight nor distributed like typical optical markers is challenging for the method.

To provide a stronger baseline, we implement our own optimization method that takes orientations and subject-specific offsets into account. The objective we minimize is $\arg\min_{\boldsymbol{\Omega}_t} \mathcal{L}_r(\boldsymbol{x}_t, \boldsymbol{\Omega}_t, \boldsymbol{O}_p)$, but to induce a prior we directly optimize in the latent space provided by VPoser [40] and add regularizers on pose and shape. The details are provided in the supplementary material. We observe that this optimization method ("pos + ori" in Tab. 1) achieves lower errors and standard deviations than MoSh++.

Learning-based We compare our method with pure learning-based approaches and train two baselines with 6 and 12 sensors respectively. The 6 sensor configuration only keeps the sensors at the wrists, lower legs, head, and back. The results are shown in Tab. 2. Both baselines take the raw measurements as inputs and map them to SMPL pose and shape with supervision on pose, shape and 3D joints. We supply subject-specific offsets $\boldsymbol{O}_p$ analogous to Sec. 5.3. Hyperparameter search was conducted for all baselines. The first baseline, *ResNet*, is a frame-wise baseline that feeds the inputs through 5 residual blocks. This is inspired by [16] who map dense marker clouds to body model parameters. The second baseline, *BiRNN*, is a bidirectional RNN adopted from DIP [17], thus modelling temporal relationships explicitly. From the results table, we can see that explicitly modelling the temporal nature of the data is helpful (the *BiRNN* outperforms the *ResNet*). We also observe that our method beats both pure learning- and optimization-based baselines. For more network and training details please refer to the supplementary material.

6.3. Ablations

Here we show the effect of major design choices on our best performing model with 12 sensors, summarized in Tab. 3. The respective results with 6 sensors are supplied in the supplementary material. We first remove the RNN which provides the initial estimate to LGD ("Ours

Model	MPJPE [mm]	PA-MPJPE [mm]	MPJAE [°]
Ours 12 no $[\boldsymbol{R}\|\boldsymbol{t}]$	167.6 ± 212.7	134.3 ± 113.3	37.5 ± 34.7
Ours 12 no $\boldsymbol{t}$	35.6 ± 25.8	29.0 ± 19.4	14.4 ± 10.0
Ours 12 ori only	50.8 ± 30.0	31.2 ± 20.4	14.3 ± 9.8
Ours 12 pos only	33.6 ± 28.3	27.5 ± 20.8	16.2 ± 11.3
Ours 12 no RNN	36.9 ± 25.4	26.5 ± 19.9	14.3 ± 10.3
Ours 12	**31.8 ± 21.0**	**24.8 ± 16.4**	**13.3 ± 9.2**

Table 3: **Ablation studies** on our best performing model.

Model	MPJPE [mm]	PA-MPJPE [mm]	MPJAE [°]
BiRNN 6	37.2 ± 26.7	33.8 ± 19.2	15.0 ± 7.8
Ours (LGD RNN) 6	**32.0 ± 25.0**	**29.5 ± 17.7**	**13.6 ± 7.3**
BiRNN 12	45.9 ± 34.3	40.2 ± 22.7	15.1 ± 8.0
Ours (LGD RNN) 12	**31.2 ± 25.7**	**24.5 ± 18.0**	**12.3 ± 7.2**

Table 4: **Cross-subject evaluation** on subject 5.

no RNN"). This architecture resembles the original, frame-wise LGD [53]. We can clearly observe the benefit of explicitly modelling the temporal nature of our data. Furthermore, we show the effect of subject-specific offsets $\boldsymbol{O}_p$ during training. The entry "no $\boldsymbol{t}$" refers to a training scheme where we set the translational part of all offsets $\boldsymbol{o}_s$ to zero and "no $[\boldsymbol{R}|\boldsymbol{t}]$" means we additionally set $\boldsymbol{R}$ to the identity. As is expected, modeling the rotational offsets has a major influence. Without these, the disparity between synthetic and real orientations is simply too large. Finally, we also experiment with feeding only position or only orientation measurements to our model ("pos/ori only"). In each case the error matched to the available modality remains reasonably low (*e.g.* "pos only" has an MPJPE of 33.6) but the respective other error increases. This justifies the choice of both modalities in our best performing model.

6.4. Cross-Subject Evaluations

LGD and our training scheme require access to subject-specific offsets. In this section we evaluate our method on an "unseen" participant whose offsets have not been used during training. To this end, we train our models with subject-specific offsets only from subjects 1-4 and hold out subject 5. Tab. 4 lists the performance of our two best models on sequences from subject 5. This again highlights the benefit of our proposed method over pure learning baselines, which is more pronounced for the 12 sensor model. This is not entirely unsurprising because LGD RNN still requires an estimate of the offsets for the iterative refinement.

6.5. Qualitative Results

We show visual comparisons of reconstructions with 6 and 12 sensors in Fig. 7. Please refer to the video and supplementary material for more visual comparisons.

Figure 7: **Visual comparisons** with 6 and 12 sensors. We show poses with self-occlusions (crouching, crossing arms) or poses that are typically challenging to recover with just 6 sensors (squatting, sitting). Images for reference only.

7. Limitations and Conclusion

Like any EM-based system, ours is susceptible to magnetic distortion due to metallic objects or other electronics that are closer than 1.5 meters to the subject. In our capture sessions we found that it is possible to control for magnetic disturbances and it also does not hinder us from capturing in everyday surroundings as shown in Fig. 7. Still, EM data can be noisy (*e.g.*, dropped frames, measurements out of calibrated range, unexpected magnetic distortion, etc.). While providing pose estimation in a noisy data regime is out of scope for this paper, we find this an interesting avenue for future work. A prototypical architecture that handles noisy inputs is described in the supplementary material. Finally, recovering detailed shape information from as little as 6 sensors is difficult as it is largely unobserved. Although there's certainly room for improvement, we see good reconstruction quality across many action types and multiple subjects. To foster future research, we release code and data.

Acknowledgments We thank Stephen Olsen and Mark Hogan for their tremendous support with the capture system. We are also very grateful for the help of Kevin Harris, Mishael Herrmann, Braden Copple, Elise Campbell, Shangchen Han, Naureen Mahmood, Thomas Langerak, Juan Zarate, Emre Aksan, and all our participants.

References

[1] Brett Allen, Brian Curless, and Zoran Popović. The space of human body shapes: Reconstruction and parameterization from range scans. *ACM Trans. Graph.*, 22(3):587–594, July 2003. 3

[2] Brett Allen, Brian Curless, Zoran Popović, and Aaron Hertzmann. Learning a correlated model of identity and pose-dependent body shape variation for real-time synthesis. In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '06, page 147–156, Goslar, DEU, 2006. Eurographics Association.

[3] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. Scape: Shape completion and animation of people. *ACM Trans. Graph.*, 24(3):408–416, July 2005. 3

[4] Andreas Aristidou, Daniel Cohen-Or, Jessica K. Hodgins, and Ariel Shamir. Self-similarity analysis for motion capture cleaning. *Comput. Graph. Forum*, 37(2):297–309, May 2018. 2

[5] Darmindra D Arumugam, Joshua D Griffin, Daniel D Stancil, and David S Ricketts. Magneto-quasistatic tracking of an american football: A goal-line measurement [measurements corner]. *IEEE Antennas and Propagation Magazine*, 55(1):138–146, 2013. 4

[6] Gabriele Bleser, Gustaf Hendeby, and Markus Miezal. Using egocentric vision to achieve robust inertial body tracking under magnetic disturbances. In *2011 10th IEEE International Symposium on Mixed and Augmented Reality*, pages 103–109, 2011. 2

[7] H. T. Butt, B. Taetz, M. Musahl, M. A. Sanchez, P. Murthy, and D. Stricker. Magnetometer robust deep human pose regression with uncertainty prediction using sparse body worn magnetic inertial measurement units. *IEEE Access*, 9:36657–36673, 2021. 2

[8] Ke-Yu Chen, Shwetak N. Patel, and Sean Keller. Finexus: Tracking precise motions of multiple fingertips using magnetic sensing. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, page 1504–1514, New York, NY, USA, 2016. Association for Computing Machinery. 3

[9] Yinfu Feng, Mingming Ji, Jin Xiao, Xiaosong Yang, Jian J. Zhang, Yueting Zhuang, and Xuelong Li. Mining spatial-temporal patterns and structural sparsity for human motion data denoising. *IEEE Transactions on Cybernetics*, 45(12):2693–2706, 2015. 2

[10] Guillermo Garcia-Hernando, Shanxin Yuan, Seungryul Baek, and Tae-Kyun Kim. First-person hand action benchmark with rgb-d videos and 3d hand pose annotations. In *Proceedings of Computer Vision and Pattern Recognition (CVPR)*, 2018. 3

[11] Andrew Gilbert, Matthew Trumble, Charles Malleson, Adrian Hilton, and John Collomosse. Fusing visual and inertial sensors with semantics for 3d human pose estimation. *International Journal of Computer Vision*, 127:1–17, 09 2018. 2

[12] Peng Guan, Alexander Weiss, Alexandru O Balan, and Michael J Black. Estimating human shape and pose from a single image. In *2009 IEEE 12th International Conference on Computer Vision*, pages 1381–1388. IEEE, 2009. 3

[13] Riza Alp Guler and Iasonas Kokkinos. Holopose: Holistic 3d human reconstruction in-the-wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10884–10894, 2019. 3

[14] 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*, pages 7297–7306, 2018. 1

[15] Nils Hasler, Hanno Ackermann, Bodo Rosenhahn, Thorsten Thormählen, and Hans-Peter Seidel. Multilinear pose and body shape estimation of dressed subjects from image sets. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1823–1830. IEEE, 2010. 3

[16] Daniel Holden. Robust solving of optical motion capture data by denoising. *ACM Trans. Graph.*, 37(4), July 2018. 2, 7

[17] Yinghao Huang, Manuel Kaufmann, Emre Aksan, Michael J. Black, Otmar Hilliges, and Gerard Pons-Moll. Deep inertial poser: Learning to reconstruct human pose from sparse inertial measurements in real time. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 37:185:1–185:15, Nov. 2018. 1, 2, 7

[18] 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. 3

[19] 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. 1

[20] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7122–7131, 2018. 3

[21] Muhammed Kocabas, Nikos Athanasiou, and Michael J. Black. Vibe: Video inference for human body pose and shape estimation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1, 7

[22] 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 *Proceedings of the IEEE International Conference on Computer Vision*, pages 2252–2261, 2019. 3

[23] Thomas Langerak, Juan Zarate, David Lindlbauer, Christian Holz, and Otmar Hilliges. Omni: Volumetric sensing and actuation of passive magnetic tools for dynamic haptic feedback. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*, UIST '20, page 594–606, New York, NY, USA, 2020. Association for Computing Machinery. 3

[24] Christoph Lassner, Javier Romero, Martin Kiefel, Federica Bogo, Michael J Black, and Peter V Gehler. Unite the people: Closing the loop between 3d and 2d human representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6050–6059, 2017. 3

[25] Lei Li, James McCann, Nancy Pollard, and Christos Faloutsos. Bolero: A principled technique for including bone length constraints in motion capture occlusion filling. In *Proceedings of the 2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '10, page 179–188, Goslar, DEU, 2010. Eurographics Association. 2

[26] T. Li, L. Fan, M. Zhao, Y. Liu, and D. Katabi. Making the invisible visible: Action recognition through walls and occlusions. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 872–881, 2019. 3

[27] Rong-Hao Liang, Kai-Yin Cheng, Chao-Huai Su, Chien-Ting Weng, Bing-Yu Chen, and De-Nian Yang. Gausssense: Attachable stylus sensing using magnetic sensor grid. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*, UIST '12, page 319–326, New York, NY, USA, 2012. Association for Computing Machinery. 3

[28] Huajun Liu, Xiaolin Wei, Jinxiang Chai, Inwoo Ha, and Taehyun Rhee. Realtime human motion control with a small number of inertial sensors. In *Symposium on Interactive 3D Graphics and Games*, pages 133–140. ACM, 2011. 2

[29] Matthew Loper, Naureen Mahmood, and Michael J Black. Mosh: Motion and shape capture from sparse markers. *ACM Transactions on Graphics (TOG)*, 33(6):220, 2014. 3

[30] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM transactions on graphics (TOG)*, 34(6):248, 2015. 2

[31] 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, Oct. 2015. 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 *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2019. 2, 3, 5, 6, 7

[33] Charles Malleson, Marco Volino, Andrew Gilbert, Matthew Trumble, John Collomosse, and Adrian Hilton. Real-time full-body motion capture from video and imus. In *2017 Fifth International Conference on 3D Vision (3DV)*, pages 449–457, 2017. 1, 2

[34] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. Vnect: Real-time 3d human pose estimation with a single rgb camera. *ACM Transactions on Graphics (TOG)*, 36(4):44, 2017. 3

[35] *Meshcapade*, accessed March 17th, 2021. https://meshcapade.com/. 4

[36] *Ascension Technology Corporation*, accessed March 9th, 2021. https://www.ndigital.com/about/ascension-technology-corporation/. 3

[37] Mohamed Omran, Christoph Lassner, Gerard Pons-Moll, Peter Gehler, and Bernt Schiele. Neural body fitting: Unifying deep learning and model based human pose and shape estimation. In *2018 International Conference on 3D Vision (3DV)*, pages 484–494. IEEE, 2018. 3

[38] *Optitrack*, accessed March 9th, 2021. https://optitrack.com/applications/movement-sciences/. 2

[39] Sang Il Park and Jessica K. Hodgins. Capturing and animating skin deformation in human motion. *ACM Trans. Graph.*, 25(3):881–889, July 2006. 3

[40] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed AA Osman, Dimitrios Tzionas, and Michael J Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10975–10985, 2019. 3, 7

[41] Maksym Perepichka, Daniel Holden, Sudhir P. Mudur, and Tiberiu Popa. Robust marker trajectory repair for mocap using kinematic reference. In *Motion, Interaction and Games*, MIG '19, New York, NY, USA, 2019. Association for Computing Machinery. 2

[42] *Polhemus Applications History*, accessed March 9th, 2021. https://polhemus.com/applications/military-old. 3

[43] *Polhemus*, accessed March 9th, 2021. https://polhemus.com. 3

[44] Gerard Pons-Moll, Andreas Baak, Juergen Gall, Laura Leal-Taixe, Meinard Mueller, Hans-Peter Seidel, and Bodo Rosenhahn. Outdoor human motion capture using inverse kinematics and von mises-fisher sampling. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1243–1250. IEEE, 2011. 2

[45] Gerard Pons-Moll, Andreas Baak, Thomas Helten, Meinard Müller, Hans-Peter Seidel, and Bodo Rosenhahn. Multisensor-fusion for 3d full-body human motion capture. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, jun 2010. 1, 2

[46] Gerard Pons-Moll, Javier Romero, Naureen Mahmood, and Michael J Black. Dyna: A model of dynamic human shape in motion. *ACM Transactions on Graphics (TOG)*, 34(4):120, 2015. 3

[47] F. H. Raab, E. B. Blood, T. O. Steiner, and H. R. Jones. Magnetic position and orientation tracking system. *IEEE Transactions on Aerospace and Electronic Systems*, AES-15(5):709–718, 1979. 3

[48] Helge Rhodin, Christian Richardt, Dan Casas, Eldar Insafutdinov, Mohammad Shafiei, Hans-Peter Seidel, Bernt Schiele, and Christian Theobalt. EgoCap: egocentric marker-less motion capture with two fisheye cameras. 35(6):162, 2016. 1, 3

[49] Daniel Roetenberg, Henk Luinge, and Per Slycke. Moven: Full 6dof human motion tracking using miniature inertial sensors. *Xsen Technologies, December*, 2007. 1, 2

[50] Daniel Roetenberg, Per Slycke, and Peter H. Veltink. Ambulatory position and orientation tracking fusing magnetic and inertial sensing. *IEEE Transactions on Biomedical Engineering*, 54(5):883–890, 2007. 3

[51] Takaaki Shiratori, Hyun Soo Park, Leonid Sigal, Yaser Sheikh, and Jessica K. Hodgins. Motion capture from body-mounted cameras. *ACM Trans. Graph.*, 30(4), July 2011. 1

[52] Leonid Sigal, Alexandru Balan, and Michael J Black. Combined discriminative and generative articulated pose and non-rigid shape estimation. In *Advances in neural informa-*

tion processing systems, pages 1337–1344, 2008. 3

[53] Jie Song, Xu Chen, and Otmar Hilliges. Human body model fitting by learned gradient descent. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX 16*, pages 744–760. Springer, 2020. 1, 2, 3, 6, 8

[54] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 529–545, 2018. 3

[55] Vince Tan, Ignas Budvytis, and Roberto Cipolla. Indirect deep structured learning for 3d human body shape and pose prediction. 2018. 3

[56] Sergio Tarantino, Francesco Clemente, D. Barone, Marco Controzzi, and Christian Cipriani. The myokinetic control interface: Tracking implanted magnets as a means for prosthetic control. *Scientific Reports*, 7, 12 2017. 3

[57] Denis Tome, Patrick Peluse, Lourdes Agapito, and Hernan Badino. xr-egopose: Egocentric 3d human pose from an hmd camera. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7728–7738, 2019. 1, 3

[58] Matthew Trumble, Andrew Gilbert, Charles Malleson, Adrian Hilton, and John Collomosse. Total capture: 3d human pose estimation fusing video and inertial sensors. In *Proceedings of 28th British Machine Vision Conference*, pages 1–13, 2017. 2

[59] Hsiao-Yu Tung, Hsiao-Wei Tung, Ersin Yumer, and Katerina Fragkiadaki. Self-supervised learning of motion capture. In *Advances in Neural Information Processing Systems*, pages 5236–5246, 2017. 1, 3

[60] Gul Varol, Duygu Ceylan, Bryan Russell, Jimei Yang, Ersin Yumer, Ivan Laptev, and Cordelia Schmid. Bodynet: Volumetric inference of 3d human body shapes. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 20–36, 2018. 3

[61] Gul Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J Black, Ivan Laptev, and Cordelia Schmid. Learning from synthetic humans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 109–117, 2017. 3

[62] *Vicon*, accessed March 9th, 2021. https://www.vicon.com/. 2

[63] Daniel Vlasic, Rolf Adelsberger, Giovanni Vannucci, John Barnwell, Markus Gross, Wojciech Matusik, and Jovan Popović. Practical motion capture in everyday surroundings. *ACM Trans. Graph.*, 26(3):35–es, July 2007. 2

[64] 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)*, sep 2018. 1, 2

[65] Timo von Marcard, Bodo Rosenhahn, Michael J Black, and Gerard Pons-Moll. Sparse inertial poser: Automatic 3D human pose estimation from sparse IMUs. In *Computer Graphics Forum*, volume 36, pages 349–360. Wiley Online Library, 2017. 1, 2, 7

[66] F. Wang, S. Zhou, S. Panev, J. Han, and D. Huang. Person-in-wifi: Fine-grained person perception using wifi. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5451–5460, 2019. 3

[67] Donglai Xiang, Hanbyul Joo, and Yaser Sheikh. Monocular total capture: Posing face, body, and hands in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10965–10974, 2019. 3

[68] Xinying Han, Hiroaki Seki, Yoshitsugu Kamiya, and Masatoshi Hikizu. Wearable handwriting input device using magnetic field. In *SICE Annual Conference 2007*, pages 365–368, 2007. 3

[69] Weipeng Xu, Avishek Chatterjee, Michael Zollhoefer, Helge Rhodin, Dushyant Mehta, Hans-Peter Seidel, and Christian Theobalt. Monoperfcap: Human performance capture from monocular video. 2018. 1, 3

[70] Yuanlu Xu, Song-Chun Zhu, and Tony Tung. Denserac: Joint 3d pose and shape estimation by dense render-and-compare. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7760–7770, 2019. 3

[71] Zhe Zhang, Chunyu Wang, Wenhu Qin, and Wenjun Zeng. Fusing wearable imus with multi-view images for human pose estimation: A geometric approach. In *CVPR*, 2020. 2

[72] M. Zhao, T. Li, M. A. Alsheikh, Y. Tian, H. Zhao, A. Torralba, and D. Katabi. Through-wall human pose estimation using radio signals. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7356–7365, 2018. 3

[73] Mingmin Zhao, Yingcheng Liu, Aniruddh Raghu, Tianhong Li, Hang Zhao, Antonio Torralba, and Dina Katabi. Through-wall human mesh recovery using radio signals. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 3

[74] Zerong Zheng, Tao Yu, Yixuan Wei, Qionghai Dai, and Yebin Liu. Deephuman: 3d human reconstruction from a single image. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7739–7749, 2019. 3

[75] Victor Brian Zordan and Nicholas C. Van Der Horst. Mapping optical motion capture data to skeletal motion using a physical model. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '03, page 245–250, Goslar, DEU, 2003. Eurographics Association. 2

Supplementary Material for EM-POSE: 3D Human Pose Estimation from Sparse Electromagnetic Trackers

Manuel Kaufmann[1,2] Yi Zhao[2] Chengcheng Tang[2] Lingling Tao[2]
Christopher Twigg[2] Jie Song[1] Robert Wang[2] Otmar Hilliges[1]

[1]ETH Zürich, Department of Computer Science [2]Facebook Reality Labs

In this supplementary material we give more details about the computation of subject-specific offsets (Sec. 1), describe training details of our proposed method (Sec. 2) and the optimization and learning baselines (Sec. 3, Sec. 4), provide details how we compute the accuracy of our EM sensors (Sec. 5), provide more quantitative comparisons (Sec. 6), show additional visualizations and failure cases (Sec. 7), describe the detailed capture protocol of our test set $\mathcal{T}$ (Sec. 8), and finally describe some initial experiments how to tackle denoising of EM data (Sec. 9). For more visualizations, please also refer to the video.

1. Computation of $\boldsymbol{O}_p$

To take into account the subject-specific offsets, we reserve a "calibration sequence" taken from $\mathcal{T}$ for each of our subjects. We use these sequences to extract per-sensor and per-subject offsets $\boldsymbol{O}_p$. The following description holds for each sensor s of subject p. We obtain $\boldsymbol{o}_s$ by first computing $\tilde{\boldsymbol{m}}_s$ on the calibration sequence following Eq. (3). This allows us to solve for $\boldsymbol{R}(t)$ and $\boldsymbol{t}(t)$ in Eq. (4) at every time step t, where we simply replace $\boldsymbol{R}_s^v$ and $\boldsymbol{p}_s^v$ with the actual real measurements.

Because $\boldsymbol{R}(t)$ and $\boldsymbol{t}(t)$ can vary over time, we extract a single estimate as follows. For $\boldsymbol{t}$ we simply compute the mean over all time steps. For $\boldsymbol{R}$ we compute the average rotation over $\boldsymbol{R}(t)$.

$$\boldsymbol{t} = \frac{1}{T}\sum_{t=1}^{T}\boldsymbol{t}(t) \tag{1}$$

$$\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T = \text{SVD}(\sum_{t=1}^{T}\boldsymbol{R}(t))$$

$$\boldsymbol{R} = \phi(\boldsymbol{U}, \boldsymbol{V}) \tag{2}$$

where ϕ extracts a valid rotation as follows:

$$\phi(\boldsymbol{U}, \boldsymbol{V}) = \boldsymbol{U} \cdot \text{diag}(1, 1, \text{sign}(\det(\boldsymbol{U}\boldsymbol{V}^T)) \cdot \boldsymbol{V}^T \tag{3}$$

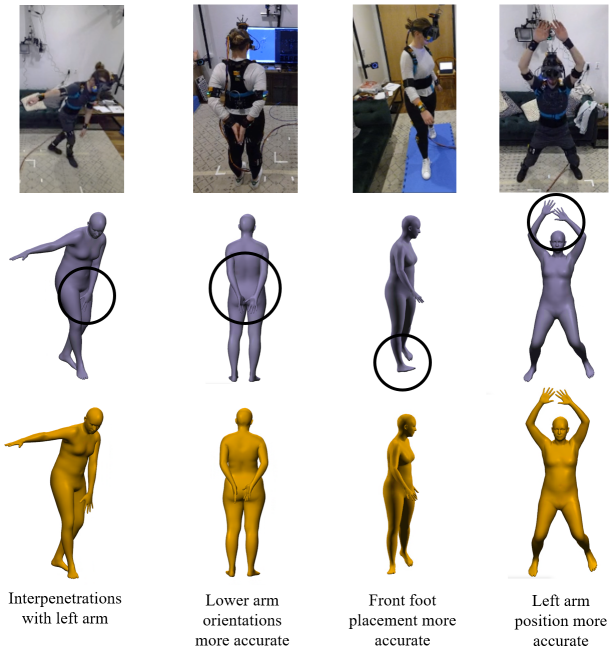

Figure 8: **Reconstructions** with 6 sensors with the BiRNN (middle row) and our best model, LGD RNN (bottom row). Differences are described directly in the figure.

2. Training Details

2.1. Normalization

We normalize our data before feeding it to our method. Since we assume that the root information is given we remove the root translation entirely by setting the SMPL root translation to zero. Then, for every sequence, we normalize the SMPL root orientation as follows (superscript n indicates it is normalized data):

$$\boldsymbol{R}_{root}^n(t) = \boldsymbol{R}_{root}^{-1}(0)\boldsymbol{R}_{root}(t)$$

Since the remaining SMPL pose parameters are all

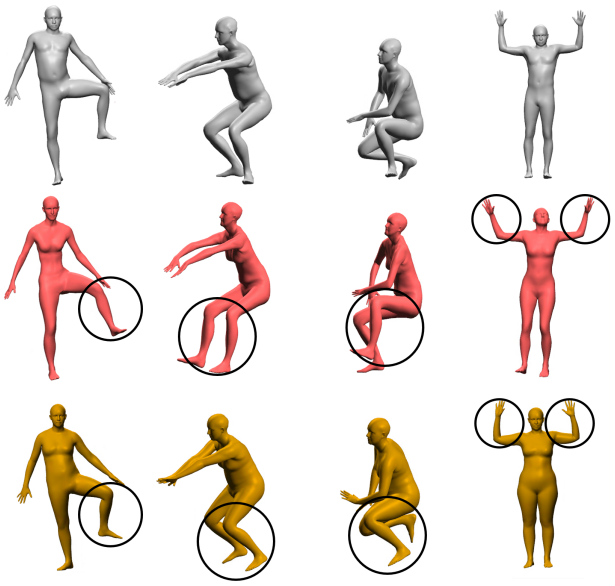

Figure 9: **MoSh++ results** [7] when using 12 sensors as input (middle row) compared to reference poses (top row) and our method (LGD RNN) with 6 sensors (bottom row).

parent-relative, this is the only normalization we perform for SMPL data. We apply the same normalization to the marker data, *i.e.*

$$\boldsymbol{p}_s^n(t) = \boldsymbol{R}_{root}^{-1}(0)(\boldsymbol{p}_s(t) - \boldsymbol{t}_{root}(t))$$
$$\boldsymbol{R}_s^n(t) = \boldsymbol{R}_{root}^{-1}(0)\boldsymbol{R}_s(t)$$

2.2. Offset augmentation

As explained in Sec. 1, the computed offsets $\boldsymbol{R}(t)$ and $\boldsymbol{t}(t)$ can vary over time. We use the translational part of the offsets to introduce some noise for data augmentation purposes during training. To do so, for every sensor and every participant we fit a multi-variate normal distribution to $\boldsymbol{t}(t)$, denoted as $\boldsymbol{\varphi}(\boldsymbol{t})$. When applying the offsets $\boldsymbol{O}_p$ as explained in Sec. 5.3 of the main paper we first draw a vector $\boldsymbol{t} \sim \boldsymbol{\varphi}(\boldsymbol{t})$ which we then use as the translational offset to obtain virtual sensors $\boldsymbol{m}_s^v$.

2.3. Architecture Details and Hyperparameters

The RNN in our proposed method, LGD RNN, consists of two LSTM layers [2] of size 512. The output of the RNN is mapped to pose and shape parameters $\boldsymbol{\Omega}_t^{(0)}$ with a dense layer. The network $\mathcal{N}$ is essentially a multi-layer perceptron (MLP). The MLP first maps its inputs, *i.e.* $\boldsymbol{\Omega}_t^{(n)}$, to the chosen hidden size, which is 512 in our case. The hidden representation is then passed to L (here 5) dense layers whereas each layer maps to the same dimensionality as the size of its inputs (*i.e.* 512). Each dense layer is preceded by a batch normalization layer [4], a PReLU activation function [1], and a dropout layer [11] in this order. The last dense layer maps back to the target dimension and thus produces the next estimate $\boldsymbol{\Omega}_t^{(n+1)}$.

We use the Adam optimizer [5] to train our models. The choice of hyperparameters are listed in Tab. 5. We use PyTorch 1.6 [8] and train all our models on a NVIDIA GeForce GTX 1080Ti, which takes roughly 16 hours.

Hyperparameter	LGD RNN 6	LGD RNN 12
α (LGD step size)	0.1	0.1
Batch size	12	12
Dropout (on inputs)	0.0	0.2
Dropout (inside MLP of $\mathcal{N}$)	0.0	0.2
λ_1 (pose loss weight)	10.0	1.0
λ_2 (shape loss weight)	1.0	1.0
λ_3 (joint loss weight)	0.1	1.0
λ_4 (reconstruction loss weight)	0.01	0.01
Learning rate	0.0005	0.0001
N (number of LGD iterations)	2	4
Number of epochs	50	50
Sequence length (training only)	32	32

Table 5: **Hyperparameters** for LGD RNN.

Hyperparameter	ResNet 6/12	BiRNN 6/12
Batch size	16	16
λ_1 (pose loss weight)	1.0	1.0
λ_2 (shape loss weight)	1.0	1.0
λ_3 (joint loss weight)	10.0	10.0
Learning rate	0.0005	0.0005
Number of epochs	50	50
Sequence length (training only)	128	128

Table 6: **Hyperparameters** for learning baselines.

3. Optimization baseline

Here we explain the details of our optimization baseline mentioned in Sec. 6.2 of the main paper. The objective function we minimize is $\arg\min_{\boldsymbol{\Omega}_t} \mathcal{L}_r(\boldsymbol{x}_t, \boldsymbol{\Omega}_t, \boldsymbol{O}_p)$, *i.e.* essentially the same objective that LGD minimizes. However, to induce a prior we operate directly in the latent space provided by VPoser [9]. This means the body parameters $\boldsymbol{\Omega}_t$ are now split into $(\boldsymbol{z}_t, \boldsymbol{\beta})$ where $\boldsymbol{z}_t$ corresponds to the latent space of VPoser. Furthermore, we add regularizers on pose and shape. The objective function we minimize thus becomes

$$\arg\min_{\boldsymbol{z}_t, \boldsymbol{\beta}} \mathcal{L}_r(\boldsymbol{x}_t, \boldsymbol{z}_t, \boldsymbol{\beta}, \boldsymbol{O}_p) + \rho_1 ||\boldsymbol{z}_t||_2^2 + \rho_2 ||\boldsymbol{\beta}||_2^2 \quad (4)$$

where we choose $\rho_1 = 10^{-6}$ and $\rho_2 = 10^{-2}$. We use PyTorch to run our optimization and use an LBFGS optimizer with a step size of 1.0 and strong Wolfe line search.

Model	MPJPE [mm]	PA-MPJPE [mm]	MPJAE [°]
Ours 6 no $\boldsymbol{t}$	44.0 ± 34.0	32.8 ± 22.3	15.8 ± 10.6
Ours 6 ori only	80.9 ± 81.4	53.5 ± 46.3	18.0 ± 13.6
Ours 6 pos only	38.6 ± 32.0	31.4 ± 25.2	17.7 ± 12.4
Ours 6 no RNN	44.4 ± 33.3	32.8 ± 23.9	16.2 ± 11.3
Ours 6	**35.4 ± 21.3**	**27.0 ± 16.3**	**14.9 ± 10.0**

Table 7: **Ablation studies** on our best performing 6-sensor model.

4. Learning baselines

Here we describe the details of our learning-based baselines, ResNet and BiRNN, as described in Sec. 6.2 of the main paper. Both baselines perform direct body parameter regression, *i.e.* we obtain SMPL pose and shape estimates $\hat{\boldsymbol{\Omega}}_t$ directly from a neural network $\nu(\boldsymbol{x}_t)$. We use the same data augmentation and preprocessing as for LGD RNN. The loss function at time step t is the same in both cases:

$$\mathcal{L}_t = \lambda_1 \mathcal{L}_1(\hat{\boldsymbol{\theta}}_t, \boldsymbol{\theta}_t^{gt}) + \lambda_2 \mathcal{L}_2(\hat{\boldsymbol{\beta}}, \boldsymbol{\beta}^{gt}) + \lambda_3 \mathcal{L}_3(\hat{\boldsymbol{J}}_t, \boldsymbol{J}_t^{gt})$$

where $\mathcal{L}_1, \mathcal{L}_3$ are the MSE and $\mathcal{L}_2$ is the L1 loss. The architectural details are explained in the following and hyperparameters are listed in Tab. 6.

ResNet The ResNet baselines is a frame-wise architecture inspired by [3]. One block consists of a dense layer that maps to the same output size as the size of the inputs, followed by a skip connection and a ReLU activation function. We use 5 such layers of dimension 1024. The output of the last layer is mapped directly to $\boldsymbol{\Omega}_t$.

BiRNN The BiRNN is a simple bidirectional RNN [10] with LSTM cells [2]. We use 2 bidirectional layers of size 256 each. The hidden forward and backward states of the last layer are mapped directly to $\boldsymbol{\Omega}_t$.

5. Computation of EM-Tracking Accuracy

To compare our EM sensors to optical marker-based tracking we glued an Optitrack rigid body to each of our sensors (*c.f.* Fig. 2 of the main paper). Here we explain in detail how we compute the disagreement between Optitrack and our sensors (*c.f.* Sec. 6.1 of the main paper). As a reminder, for every sensor s and every time step t we obtain four measurements: the Optitrack 6D pose, *i.e.* $\boldsymbol{p}_s^O(t)$ and $\boldsymbol{R}_s^O(t)$, and the EM 6D pose, *i.e.* $\boldsymbol{p}_s^M(t)$ and $\boldsymbol{R}_s^M(t)$. All measurements are calibrated to world space and hence, under perfect agreement, a constant rigid transformation $[\boldsymbol{R} \mid \boldsymbol{t}]$ would relate the two. We characterize the agreement by computing this rigid transformation and measuring how much it changes over time as follows. For the positional agreement $e_s^{pos}(t)$ we simply compute the deviation from the mean translational offset.

$$\boldsymbol{t} = \frac{1}{T} \sum_{t=1}^{T} \boldsymbol{p}_s^M(t) - \boldsymbol{p}_s^O(t)$$
$$e_s^{pos}(t) = ||\boldsymbol{p}_s^M(t) - \boldsymbol{p}_s^O(t) - \boldsymbol{t}||_2$$

For the angular error $e_s^{ang}(t)$ we proceed similarly and solve an orthogonal Procrustes problem to find the constant rotation $\boldsymbol{R}$ that best relates $\boldsymbol{R}_s^M$ and $\boldsymbol{R}_s^O$ as follows:

$$\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T = \text{SVD}\left(\sum_{t=1}^{T} (\boldsymbol{R}_s^M(t))^T \boldsymbol{R}_s^O(t)\right)$$
$$\boldsymbol{R} = \phi(\boldsymbol{U}, \boldsymbol{V})$$
$$e_s^{ang}(t) = \text{dist}(\boldsymbol{R}_s^M(t)\boldsymbol{R}, \boldsymbol{R}_s^O(t))$$

where ϕ is defined in Eq. (3) and $\text{dist}(\cdot)$ finds the closest angle of rotation between its inputs. To do so we first convert the rotation matrices to quaternions and then use:

$$\text{dist}(\boldsymbol{q}_1, \boldsymbol{q}_2) = \cos^{-1}\left(2\langle \boldsymbol{q}_1(t), \boldsymbol{q}_2(t)\rangle^2 - 1\right)$$

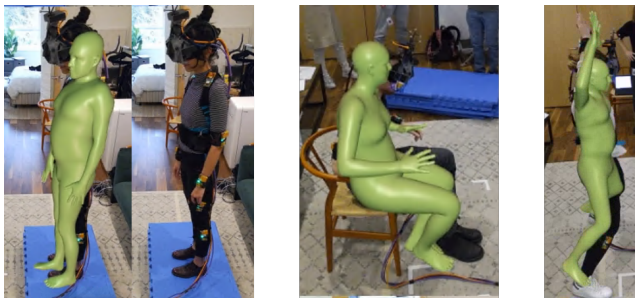

Figure 10: **Failure cases.** Inaccurate shape reconstruction especially around abdomen (left) and challenging lower leg orientations (middle and right).

6. More Quantitative Results

6.1. Comparison to RGB Methods

We compare our method to a state-of-the-art monocular RGB-based pose estimator, VIBE [6]. To do so, we select the camera facing the front of the subject as input to VIBE. The results are shown in Tab. 8. The error is only computed on frames for which VIBE detected a person. As we can see, VIBE does not perform favourably on our data. This is not unexpected because a) our imagery is a real in-the-wild scenario and b) the inputs to VIBE and our method are vastly different. Under these circumstances, VIBE still performs admirably. We see this experiment as further motivation to employ EM-based systems to gather reference data to boost RGB-based methods down the line.

Model	MPJPE [mm]	PA-MPJPE [mm]	MPJAE [°]
VIBE [6]	100.3 ± 79.3	70.1 ± 58.2	24.8 ± 15.7
Ours (LGD-RNN) 6	**36.4 ± 23.7**	**28.1 ± 16.9**	**13.6 ± 8.8**

Table 8: **Comparison to VIBE** on 10 representative test sequences from subjects 1-4.

6.2. Ablation Study with 6 Sensors

In Tab. 7 we provide the same ablation study as in Sec. 6.3 of the main paper but with the 6 instead of the 12 sensor model. Based on this table we can see that the same conclusions hold as already drawn in the main paper.

7. More Visualizations

To highlight the differences between our best model, LGD RNN, and its closest baseline, BiRNN, we compare their performance visually in Fig. 8. We can see that the BiRNN sometimes produces interpenetrations and lacks some accuracy at the end effectors.

Furthermore, we also compare the performance of MoSh++ [7] in Fig. 9. The chosen frames highlight that the shape estimates of MoSh++ are sometimes off by quite a margin. This is because it makes different assumptions about the sensor-to-skin offsets. Furthermore, the orientation of end effector segments often exhibit a high error in the MoSh++ results. This is not unexpected since it only uses 12 positional estimates. In contrast, our best model (*c.f.* bottom row in Fig. 9) produces more accurate limb orientations even with only 6 sensors as it uses both position *and* orientation inputs.

We also show failure cases of our method in Fig. 10. Shape estimation from just a few on-skin measurements is challenging. We sometimes see bulging bellies (*c.f.* Fig. 10) and inaccurate shape in the hip region (*c.f.* bottom right corner in Fig. 9). Getting the lower leg orientation correct in extreme articulations is difficult, too, even with explicit orientation measurements (*c.f.* Fig. 10). In addition, such errors are visually very striking as they can cause foot sliding.

8. Test Set Details

We describe the detailed content and capture protocol of our test set $\mathcal{T}$ in Tab. 9. All participants were guided by an assistant, participated voluntarily and gave written consent to record and publish their data.

9. Denoising Experiments

The data measured by our EM-based capture system can be noisy. Typical sources of noise include dropped frames (due to sensor malfunctions or wireless connections), increased jitter when operating outside the calibrated range, or unexpected magnetic distortion. Our proposed method, LGD RNN, iteratively fits SMPL to the observed EM data. Hence, it is clear that LGD RNN cannot handle certain types of noise, such as dropped sensors. We find pose estima-

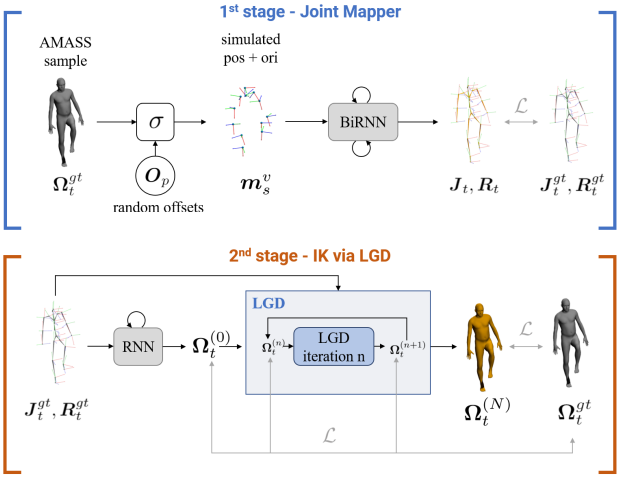

Figure 11: **Denoising architecture overview.** Our architecture to estimate SMPL pose and shape with noisy input measurements works in two stages. The first stage, called joint mapper, maps EM data to 3D SMPL joints $\boldsymbol{J}_t$ and root-relative joint orientations $\boldsymbol{R}_t$. This is a simple two-layer BiRNN which we directly supervise with the ground-truth joint positions and orientations. We randomly remove one or several sensor measurements from the input for half the duration of the given sequence. The second stage is performing IK and uses the LGD framework to do so. The orientations $\boldsymbol{R}_t$ help to disambiguate the orientation of bone segments (especially so for end effectors). At training time we use synthetic EM measurements $\boldsymbol{m}_s^v$ to train the joint mapper. Ground-truth joint positions $\boldsymbol{J}_t^{gt}$ and orientations $\boldsymbol{R}_t^{gt}$ extracted from AMASS are used to train the IK stage. At test time we simply feed the real EM data to the joint mapper and the output of the joint mapper to the IK stage.

Figure 12: **Denoising comparison.** For this frame, the right lower leg sensor is missing in the input. LGD RNN struggles to reconstruct the pose (middle) whereas the two-stage approach does a better job (right). Notice how the missing sensor is affecting the entire pose output for LGD RNN.

Action Type	Description	# Frames	Minutes
Arms ROM	Arm raises, arm swings, cross arms, clap hands front and back with straight arms.	14,720	8.2
Arms Fast	Fast arm swings, pretend to play Beat Saber VR, punches, rotate wrists around each other fast.	7,169	4.0
Calibration	Move head left to right and rotate wrists in T-Pose, move head left to right and rotat rotate wrists when arms stretched in front, one leg raise each.	3,709	2.1
Head and Shoulders	Nod head back and forth, move head left to right, roll head left to right, rotate shoulders forwards and backwards, rotate torso, bend over and move arms around.	9,498	5.3
Jumping Jacks	3-5 jumping jacks.	1,952	1.1
Lower Body	Leg raises left and right, raise leg then rotate outwards, squats, crouching	7,305	4.1
Lunges	Crouching, several lunges with left and right foot in front.	4,714	2.6
Sitting on chair	Grab a chair, sit on chair, move one leg over the other, pretend to sit at a table and interact with PC, keyboard, touch screens.	9,362	5.2
Walking	Walk normally from left to right in capture area, side-stepping from left to right with and without crossing over the legs.	8,391	4.7
Total		**66,820**	**37.1**

Table 9: **Test set $\mathcal{T}$.** Description and length of the sequences in our test set $\mathcal{T}$. Each of the 5 participants performed the actions described here in a single session. Each sequence starts and ends with a T-Pose.

Frames	Model	MPJPE [mm]	PA-MPJPE [mm]	MPJAE [°]
all	LGD RNN 12	**33.3** ± 24.3	**26.2** ± 18.8	**13.4** ± 9.2
	2-stage 12	38.8 ± **22.0**	28.2 ± **17.3**	13.8 ± **9.1**
avail.	LGD RNN 12	**31.8** ± **21.0**	**24.8** ± **16.4**	**13.3** ± 9.2
	2-stage 12	39.1 ± 21.3	28.1 ± 16.7	13.7 ± **9.1**
miss.	LGD RNN 12	45.6 ± 40.6	37.7 ± 30.0	15.0 ± 9.3
	2-stage 12	**36.2** ± **27.0**	**29.1** ± **21.9**	**14.6** ± **9.1**

Table 10: **Denoising experiments** on all frames (*all*), only on frames with*out* missing sensors (*avail.*) and only on frames with at least one missing sensor (*miss.*).

tion in such a noisy data regime an interesting direction for future work and experimented with an initial architecture that can cope with dropped frames and magnetic distortion which we briefly describe in the following.

Although LGD RNN *can* handle some noise (by means of incorporating pose priors), it is not meant to be a denoising architecture per se as it fits SMPL pose and shape to the inputs directly. Hence, our idea is to separate the tasks of denoising and fitting into separate modules and came up with a two-stage architecture. The first stage, also called joint mapper, regresses SMPL 3D joint positions and root-relative joint orientations from the input observations. In this stage we randomly remove sensors from the input to simulate dropped frames. Thus, the joint mapper maps to a proxy representation that is close to SMPL while also denoising the inputs. The second stage then lifts the output of the joint mapper to the final estimate of SMPL pose and shape. This is again an LGD-based iterative fitting procedure. Experimentally we have found that using LGD for this stage outperforms an optimization-based IK step. Both stages are trained independently. For an overview, please refer to Fig. 11.

In our experiments we have found that this two-stage architecture yields good results. The final SMPL poses are smooth and with 12 input sensors 1-2 missing sensors are compensated plausibly. This is also reflected in quantitative comparisons shown in Tab. 10. In this table we compare the two-stage approach to LGD RNN when using 12 sensors. We report its performance on 3 sets of frames: frames that have no missing sensors (*avail.*), frames with at least one missing sensor (*miss.*) and the union of these two sets, which corresponds to all frames in our test set (*all*). We observe that the two-stage approach does not beat LGD RNN on the good frames where no sensor data is missing, but remains competitive. It also trails behind LGD RNN on all the frames, which makes sense since we have many more "good" frames than frames with missing sensors. However, on frames where at least one sensor is missing (*miss.* in Tab. 10), the two-stage architecture shows its potential and clearly outperforms LGD RNN. For a visual example of a denoised frame please refer to Fig. 12.

The two-stage model is not only interesting to fill in missing sensor data. If we assume we have a mechanism to detect magnetic distortion, we can simply suppress the sensor measurement for those time instances where magnetic

distortion is detected. We can then use the same two-stage architecture to remedy the impact of EM interference. We find this an interesting direction for exploration and release code and data to foster future research.

References

[1] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1026–1034, 2015. 2

[2] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 2, 3

[3] Daniel Holden. Robust solving of optical motion capture data by denoising. *ACM Trans. Graph.*, 37(4), July 2018. 3

[4] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, page 448–456. JMLR.org, 2015. 2

[5] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2015. 2

[6] Muhammed Kocabas, Nikos Athanasiou, and Michael J. Black. Vibe: Video inference for human body pose and shape estimation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 3, 4

[7] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. Amass: Archive of motion capture as surface shapes. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2019. 2, 4

[8] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. 2

[9] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed AA Osman, Dimitrios Tzionas, and Michael J Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10975–10985, 2019. 2

[10] Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681, 1997. 3

[11] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014. 2