Mobile, Egocentric Human Body Motion Reconstruction Using Only Eyeglasses-mounted Cameras and a Few Body-worn Inertial Sensors

Young-Woon Cha*, Husam Shaik*, Qian Zhang*, Fan Feng*, Andrei State*,§, Adrian Ilie*, Henry Fuchs *

*Department of Computer Science, University of North Carolina at Chapel Hill
§InnerOptic Technology, Inc.

Figure 1: Mobile, egocentric real-time body motion capture system using only eyeglasses-mounted cameras and a few body-worn inertial sensors. 2x2 image groups at left and center: fast body motion reconstructions of indoor and outdoor user (top), shown in VR (bottom). Right: current mobile user (color), and future vision (monochrome) depicting casual everyday use of streamlined system with miniaturized cameras embedded in the frames of wide-field-of-view AR eyeglasses, and IMUs on wrists and in shoes.

Abstract

We envision a convenient telepresence system available to users anywhere, anytime. Such a system requires displays and sensors embedded in commonly worn items such as eyeglasses, wristwatches, and shoes. To that end, we present a standalone real-time system for the dynamic 3D capture of a person, relying only on cameras embedded into a head-worn device, and on Inertial Measurement Units (IMUs) worn on the wrists and ankles. Our prototype system egocentrically reconstructs the wearer's motion via learning-based pose estimation, which fuses inputs from visual and inertial sensors that complement each other, overcoming challenges such as inconsistent limb visibility in head-worn views, as well as pose ambiguity from sparse IMUs. The estimated pose is continuously re-targeted to a prescanned surface model, resulting in a high-fidelity 3D reconstruction. We demonstrate our system by reconstructing various human body movements and show that our visual-inertial learning-based method, which runs in real time, outperforms both visual-only and inertial-only approaches. We captured an egocentric visual-inertial 3D human pose dataset publicly available at `https://sites.google.com/site/youngwooncha/egovip` for training and evaluating similar methods.

Index Terms: Computing methodologies—Computer graphics—Graphics systems and interfaces—Virtual reality; Computing methodologies—Computer graphics—Animation—Motion capture; Computing methodologies—Artificial intelligence—Computer vision—Reconstruction; Computing methodologies—Machine learning—Machine learning approaches—Neural networks

*e-mail:{youngcha,hshaik,qzane,fan8,andrei,adyilie,fuchs}@cs.unc.edu

1 Introduction

Telepresence enables remote social interaction without physical presence. 3D display greatly enhances the sense of presence, but requires the ability to fully capture and reconstruct human subjects as well as their environment.

We expect 3D capture of user experiences to become a feature of common head-worn devices in the near future. Today's ubiquitous mobile phones and augmented reality (AR) systems such as the Microsoft HoloLens [23] may eventually evolve into the form factor of conventional eyeglasses, with transparent see-through and wide-field-of-view (FoV) capabilities, to be worn all day like ordinary eyeglasses. With widely available wearable technology embedded in commonly worn accessories (cameras in eyeglasses, IMUs in wristwatches and shoes), a mobile 3D acquisition and display system such as the one in Fig. 1 (right) will enable 3D telepresence.

One of the challenges of targeting an eyeglass-frame form factor is that the user's limb motions are frequently unobservable by the cameras due to occlusion, or to being outside of the camera views, as illustrated in Fig. 2 and Fig. 3. This problem makes many prior pose estimation methods inapplicable to situations like ours. For example, per-frame visual 3D pose estimation methods can produce unreliable estimates for occluded joints [9] due to incomplete visi-

bility. Similarly, while human performance capture approaches that use external cameras have achieved high accuracy and real-time performance [12, 13, 16, 43], they require all joints to be visible. Joint heatmap estimation methods [6, 24, 47] are also unable to handle the joints that are outside the image because they cannot be labeled within the 2D heatmap. Extending the heatmap size by padding the boundary is likely to generate high 3D joint errors due to the high distortion of wide-FoV or fisheye lenses. Finally, prior egocentric capture headgear [7, 28, 33, 45] featured cameras mounted farther away from the face; while they offer better body and limb visibility, they are obtrusive and thus unacceptable for daily use.

Another challenge is reducing the number of IMU sensors for widespread acceptability. Prior visual-inertial fusion approaches for 3D pose estimation [18, 39, 40] require more than 10 body-worn sensors, a number unlikely to be accepted for general use, even with miniaturization. Reducing that number results in pose ambiguities and lower accuracy for non-instrumented body parts [14, 32, 41]. For example, a knee raise cannot be reliably distinguished from a standing pose, as the IMU data is insufficient for inferring thigh orientation if no sensor is worn on it.

In this paper, we present a wearable 3D acquisition system for real-time 3D mobile telepresence relying only on eyeglass-frame-mounted cameras and IMUs on wrists and ankles. This approach allows for convenient, unobtrusive reconstruction and communication of experiences at any indoor or outdoor location. To support the vision of such a fully mobile capture system, we capture the wearer's 3D body pose using learning-based visual-inertial sensor fusion. Unlike methods that rely on instrumented environments [12, 46], this enables completely self-contained egocentric content capture and overcomes inconsistent limb visibility, as well as IMU pose ambiguity caused by sparse IMUs.

Our approach consists of three components which allow visual and inertial measurements to complement each other when tracking joints. First, a *visibility-aware visual 3D pose network* estimates visible 3D joints while suppressing unreliably detected occluded joints. Second, an *online IMU offset calibration method* improves the inertial measurements by aligning the visual and inertial bone orientations, over time, for forearms and lower legs with attached IMUs. Third, a *visual-inertial 3D pose network* estimates the poses of upper arms and thighs without IMUs by using a sequence of inertial measurements of the corresponding lower bones, as well as visual detection of the upper bones in previous frames. At each instant, the estimated body pose is re-targeted to a human surface model, resulting in a high-fidelity reconstruction of the user. The full body pose, including 3D joint locations as well as 3D bone orientations, is estimated continuously and kept temporally coherent, even when some joints are out of image or occluded.

We demonstrate our system on reconstructions of various human body movements in a remotely assisted physical therapy scenario and show the mobile capability in an outdoor scenario. For training and evaluation, we collected a new large-scale egocentric visual-inertial 3D human pose dataset. We know of no existing dataset that includes occlusion, out-of-image labels in egocentric views, and densely worn inertial sensors. We plan to make our dataset publicly available. In experiments, our visual-inertial learning-based method runs in real-time, at 30 Hz, on a standard PC, and outperforms both visual-only and inertial-only approaches, showing significant improvements in out-of-image and self-occlusion situations.

Our main contributions are:

- The first egocentric 3D human pose estimation approach that can handle both sparse visibility and sparse inertial sensors.
- A working, standalone, proof-of-concept prototype in an eyeglasses form factor for mobile capture and real-time body motion estimation.
- The first egocentric human motion dataset that includes multiple views with joint visibility information as well as inertial measurements.

2 Related Work

2.1 Body Reconstruction

Deformable body model-based surface estimation has been a focus in computer vision [17]. Estimation of model parameters approximates the human surface in conjunction with visual pose estimation [4], by estimating dense correspondences between the body model and imagery [2], or by direct volumetric inference [36]. Recent work shows advances in real-time performance by using temporal poses [16], as well as face and hand poses [43]. High-fidelity geometry can also be estimated by fitting image silhouettes [12], or by cloth simulation [46]. These approaches require external camera views to be able to fit full body shapes and poses, and assume full body visibility in the images. In egocentric views, however, body parts are often invisible.

2.2 Visual Pose Estimation

Recent advances in learning-based approaches for deep neural networks have shown significant improvements in accuracy when used for pose estimation. 2D joint heatmap-based estimation has been successful using Convolutional Neural Network (CNN) architectures [6, 24, 47]. CNN-based 3D joint estimations also have shown significant accuracy in real time for a single outside-in looking view [21, 22]. Human pose constraints [10, 31] and occlusion information [9] have been incorporated during training. In the case of continuous human motions over time, Recurrent Neural Network (RNN)-based pose estimations have shown promising results for a sequence of motion predictions [5, 19, 38]. These approaches estimate joint locations, but 3D bone orientation estimation is still an open problem when using only visual information to estimate a full body pose.

2.3 Visual Egocentric Pose Estimation

High-quality reconstruction from egocentric data captured by body-worn cameras remains a challenge, requiring reconstruction methods that operate in arbitrary, uninstrumented environments. Outside-looking-in camera-based human pose estimation methods are not directly applicable to egocentric views of the body.

Body motion can be inferred from egocentric body-worn cameras using structure-from-motion [29] or learning-based approaches [8, 15]. However, without direct observation of the body, pose estimation accuracy is limited. Significant improvements have been made using head-worn, downward-looking wide-FoV cameras, which enable views of most of the wearer's body [7, 28, 33, 45]. Learning-based approaches have been proposed to deal with the unusual viewpoints. Recent methods based on a single head-worn camera view [33,45] have used less-obtrusively mounted cameras to arrive at pose estimation improvements. However, the form factors employed are still too obtrusive for wide acceptability.

Approaches using downward near-body views have yet to fully address the challenges of self-occlusion and out-of-view joints, which need to be resolved in order to estimate a full body pose of the wearer solely from body-worn cameras.

2.4 Inertial Pose Estimation

Human pose estimation can also be performed using body-worn inertial measurement units (IMUs). IMUs can capture fast motions [18] and track body parts that might be occluded in camera views, but they suffer from measurement noise and drift over time, and require careful calibration for the initial pose.

Even with miniaturization of sensors, using a relatively large number of worn sensors is unlikely to be widely accepted. To increase acceptability, recent approaches have attempted to reduce the number of IMUs to a sparse set by employing temporal orientations and accelerations [14, 32, 41]. The IMUs are worn only on forearm and lower leg; the missing upper arm and thigh orientations are estimated by assuming that the temporal motions of lower and

upper bones are highly correlated. Inference results are promising, but suffer from pose ambiguity, as multiple poses can be possible with similar measurements. This issue is addressed only partially by using more temporal measurements such as future frames or an entire sequence. To overcome this problem, visual and inertial sensor fusion [18, 34, 39, 40] leverages outside-looking-in cameras jointly with IMUs to calculate a 3D body pose. Visual pose estimates from the outside-looking-in cameras help constrain the possible 3D poses of the inertial sensors, and alleviate the IMU measurement noise [39]. However, so far these approaches require complete body visibility, which is seldom achievable from egocentric views.

3 Wearable Capture and Egocentric Dataset

3.1 Eyeglasses and IMUs Prototype

Our goal is to develop a fully mobile telepresence system whose sensors are embedded in commonly worn items such as eyeglasses, wristbands, and shoes. Toward that end, our current prototype uses cameras in eyeglasses frames and only 4 IMUs (Xsens MTw Awinda on wrists and ankles). Adding more IMUs (e.g., on the torso, elbows and knees) improves the results, but the added inconvenience would considerably reduce acceptability. As shown in Sect. 5, the combination of multiple cameras, 4 IMUs and deep learning-based techniques is sufficient to fill in the "missing" sensor data from elbows and knees.

We envision a headset design (shown in Fig. 2d) with 4 miniature cameras: 2 downward-looking cameras placed at the bottom outside corners of the frame to observe the user's body, and 2 forward-looking cameras placed at the top outside corners of the frame to observe the environment. Compared to previous egocentric headsets [7, 28, 33, 45], our design is more user-friendly, but makes the 2 downward-looking viewpoints significantly more challenging as body parts are frequently out of view or occluded.

Working towards this design, we have built a preliminary prototype with available larger cameras (Toshiba Teli BU505MCF), mounted on a 3D-printed eyeglasses frame as shown in Fig. 2a. We currently use only 3 cameras (two 160°FoV downward-looking cameras; one 121°FoV forward-looking camera).

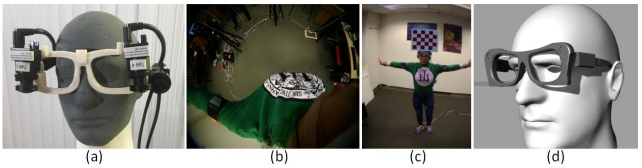

Figure 2: (a) Current headset capture prototype. (b) T-pose in downward camera, viewpoint is worse than in prior egocentric setups [33, 45]. (c) T-pose from external viewpoint. (d) Future eyeglass-form factor design.

3.2 Egocentric Visual+Inertial Human Pose Dataset

Following recent work in egocentric video and IMU-based pose estimation, we decided on a learning-based approach to use with our prototype. However, none of the available egocentric datasets were suitable for training, because their viewpoints are farther away from the user's face, they contain no visibility information, and they are monocular. We could not use existing IMU datasets either, as they were lacking accompanying egocentric video data. Consequently, we collected a new human pose dataset with users wearing our prototype headset and 8 IMUs. The ground truth full-body 3D joints are acquired using multiple wall-mounted cameras in a capture studio [7]. We recorded various types of motions for multiple users, including normal-speed as well as high-speed actions such as walking, sitting, gesturing, running, and physical therapy. A few examples are shown in Fig. 3.

We collected 22 sequences for training and 9 sequences for evaluation with 6 human subjects, for a total of 38k frames of visual+inertial data. The summary of the dataset is shown in Table 1.

Figure 3: Incomplete body visibility in eyeglass form factor views. Top row: Selected head-worn views from our Egocentric Visual Inertial Pose Dataset (Ego-VIP) with labeled visibility information. Bottom row: Corresponding external views in reference data.

For the visual training data, 11k real images were uniformly sampled and manually filtered from the full recording. 38k synthetic images were generated using the body pose from the real data with the following random augmentations [33, 45]: clothing and background texture, head rotation, and headgear translation. Each joint visibility was estimated using the z-buffer of the projected body model onto the egocentric image and labeled as visible, occluded, or outside the FoV. Torso joints (neck, shoulders, and hips) were labeled as visible regardless of occlusion because they play an essential role as root joints for bone pose estimation.

The inertial data from the 8 sensors was synchronized with the visual data and calibrated using the method in Sec. 4.4. 38k frames of real IMU data were augmented by mirroring the pose front-to-back and side-to-side, temporally smoothing pose orientations, and introducing random acceleration noise.

Table 1: Egocentric Visual-Inertial Pose Dataset (Ego-VIP), in number of frames.

	Real Size	Synthetic Size	Training Size	Test Size
Visual	11,822	38,588	50,410	13,213
Inertial	38,971	350,739	389,710	13,213

To the best of our knowledge, this is the first dataset that includes stereo egocentric views with joint visibility and calibrated inertial data. The joint visibility information is crucial for training occlusion-aware joint detectors.

4 Egocentric Reconstruction Method

Working toward our goal of fully mobile telepresence, we devised a real-time full body shape and pose reconstruction method using only egocentric devices we deem convenient and acceptable for daily wear: eyeglasses-mounted cameras and a few body-worn IMUs. The available information from the visual-inertial sensors is too sparse for each sensing modality to estimate the full body pose by itself. First, limb motions are frequently occluded by the body or are invisible due to being outside the camera views. Second, IMUs are worn only on forearms and lower legs, so upper arm and thigh orientations are missing. To solve this ill-constrained problem, we employed a *visibility-aware visual pose network* and a *temporally-integrated visual and inertial pose network*. The 3D reconstruction pipeline is illustrated in Fig. 4. It consists of three main stages.

In the first stage, a *visibility-aware 3D joint detector network* (Sect. 4.2) estimates the 3D positions of joints observable in the two egocentric downward views. The detected 3D joints are transformed to world space (Sect. 4.3) using the headset pose estimated via *VSLAM* [30].

In the second stage, the 3D orientations of lower bones (forearms, lower legs) and upper bones (upper arms, thighs) are estimated using a *visual-inertial IMU offset calibrator* (Sect. 4.4) and a *temporal visual-inertial orientation network* (Sect. 4.5), respectively.

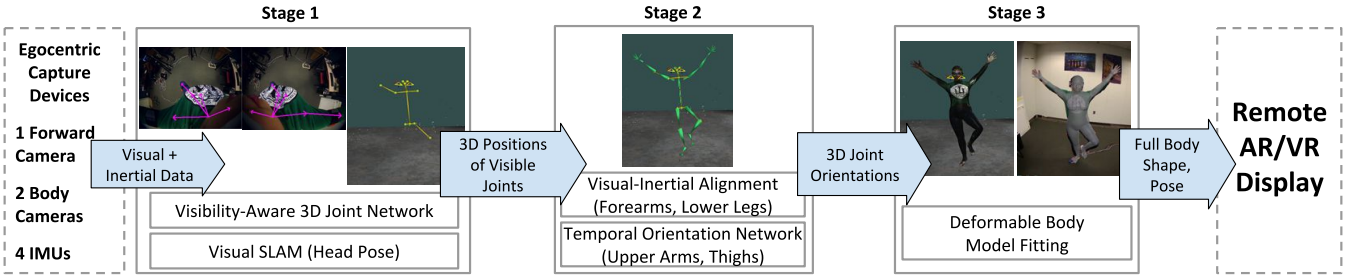

Figure 4: 3D reconstruction pipeline.

In the third stage (Sect. 4.6), the shape and pose of the parametric body model are estimated using the estimated full-body 3D joint locations and orientations from the second stage.

4.1 3D Body Representation

We use the *SMPL* parametric body model [17] to represent the body shape and pose. It consists of 10 shape parameters β and $24 \cdot 3 = 72$ pose parameters θ, which deform a triangular mesh $\mathcal{M}(\theta, \beta)$ with 6480 vertices using linear blend skinning.

Instead of representing θ as a set of local bone rotations, we use the equivalent bone representation defined as a set of global transforms $T^M \in \mathbb{R}^{4\times4}$. We use ^{M} to denote the body *M*esh space and ^{S} to denote the *S*keleton space. In this representation, a bone i is defined by two connected joints and a transform (Fig. 5).

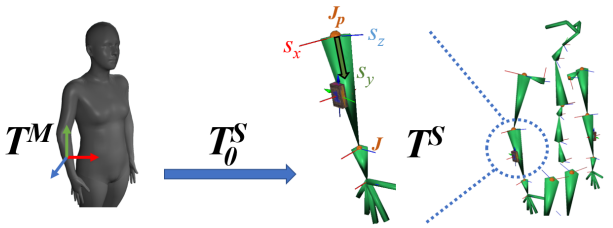

Figure 5: Bone representation. A bone (forearm) consists of a base joint (elbow) J_p, a tip joint (wrist) J, and an orientation $R^S = [s_x, s_y, s_z]$. They form a bone transformation T^S in the skeleton. The pose parameter T^M in the 3D mesh can be converted into skeleton space using the bind pose matrix T_0^S from the rest pose.

We define the skeletal bone transformation $T_i^S \in \mathbb{R}^{4\times4}$ in global space as a convenient way to represent the pose in the skeleton as:

$$T_i^S = \begin{bmatrix} R_i^S & J_{p(i)} \\ \mathbf{0} & 1 \end{bmatrix} \tag{1}$$

$R^S = [s_x, s_y, s_z] \in \mathbb{R}^{3\times3}$ is the bone rotation and J_p is the base joint position. The column vectors of R^S form the 3D axes of the bone and the axis $s_y = R^{[:,2]}$ represents the bone direction d_i from the base (parent) to tip (child) joint: $d_i = (J_i - J_{p(i)})/(||J_i - J_{p(i)}||)$. We denote the bone direction computed from a rotation as:

$$d_i = d(R_i) = R_i^{[:,2]} \tag{2}$$

The pose parameter T_i^M can be directly computed from T_i^S as:

$$T_i^M = T_i^S (T_{i,0}^S)^{-1} \tag{3}$$

The bind pose matrix $T_{i,0}^S$ maps the coordinate frames $\mathcal{F}^M \mapsto \mathcal{F}^S$, is calculated using the joint positions in the rest pose of the body model, and updated only when the shape parameters β are changed. In the rest pose, T_i^M is the identity matrix.

The joint positions in rest pose J_0 are described by the joint regressor $\mathcal{J}$ from the shaped vertices. We estimate the body shape β using the unposed joints $J_0 = (T^M)^{-1}(J)$ by minimizing E_{shape}:

$$E_{shape} = \sum_{i=1}^{K} ||(T_i^M)^{-1}(J_i) - \mathcal{J}_i(\mathcal{M}_0 + \mathcal{B}_s(\beta))||_2^2 + w_s||\beta||_2^2 \tag{4}$$

$w_s = 0.001$ is a weight for the regularization term, and $K = 13$ is the number of joints. The vertices are reshaped by the mean shape $\mathcal{M}_0$ and the linear blend shapes $\mathcal{B}_s(\beta)$.

4.2 Visibility-Aware 3D Joint Detection Network

In visual human pose estimation, occluded joints often lead to erroneous results [9]. When using egocentric images, legs are frequently occluded by the body, and arms can be out of camera FoV [7, 45]. Our visibility-aware 3D joint detection network takes a $m \times m$ egocentric image as input ($m = 320$) and estimates only the observable joints while rejecting unreliable joints by incorporating joint visibility information. The egocentric dataset described in Sect. 3.2 is labeled with visibility information, enabling visibility awareness training. The *g*round *t*ruth (^{gt}) binary *v*isibility v^{gt} is set to 1 for visible joints and 0 for invisible (occluded or outside of FoV) joints.

We extend the *Stacked Hourglass* architecture [24] used in 2D human pose estimation to a 3D joint estimation network (Fig. 6). In a head-worn wide-FoV camera image, lower body joints appear significantly smaller than upper body joints. Instead of using multi-scale images [45], we take advantage of the fact that the Hourglass module inherently collects information across all image scales. We also use a *DSNT* regression module [25] to estimate 2D coordinates from heatmaps. This increases computational efficiency, as heatmaps no longer need to be transferred to the CPU for parsing at runtime.

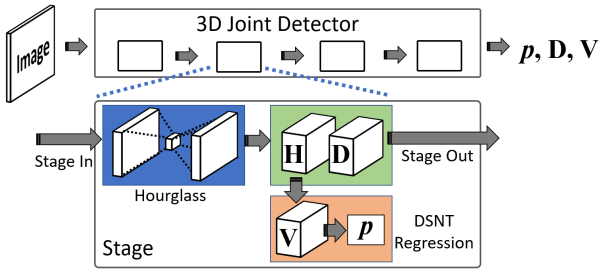

Figure 6: Network Structure for the 3D Joint Detector. The Hourglass module outputs joint heatmaps H and depthmaps D as concatenated channels. H and D are propagated into the next stage. The regression module outputs 2D coordinates p from confidence maps V normalized by H. Given a single input image, the 4 stage-network outputs p, D, V, from which 3D joint coordinates are computed.

The Hourglass module infers heatmaps $H \in \mathbb{R}^{(m/4)\times(m/4)\times K}$ in the first K channels and inverse depthmaps $D \in \mathbb{R}^{(m/4)\times(m/4)\times K}$ in the last K channels. H are normalized into confidence maps V by a Softmax layer. V are transformed into 2D coordinates $\mathbf{p}$ by the dot product of the X- and Y-coordinate matrices [25]. The inverse depthmap D is a heatmap containing normalized inverse depth values for joints. The normalized inverse depth value is defined as $(d_{max} - d)/d_{max}$, where d is a depth in meters and $d_{max} = 2$ is the maximum depth. Distances close to the camera are assigned higher values, and farther distances are assigned near-zero values [42].

Confidence $\tilde{v}$ and depth d are read out at the estimated $\mathbf{p} = (x, y)$ coordinate in V and D, respectively. When confidence $\tilde{v}$ is large enough ($\tilde{v} > t_v$, with $t_v = 0.05$), coordinate $\mathbf{p}$ is considered valid and visibility v is set to 1, otherwise it is set to 0. The raw inverse depth read-out is transformed back into depth d in meters. The 3D joint position is computed by back-projecting (x, y, d) using the camera calibration matrix. The concatenated H and D, as the output of the stage, are propagated into the next stage as input. We use 4 stacked stages, taking into account both accuracy and speed.

The network is trained to minimize the loss function $\mathscr{L}_{joint\_net} = \mathscr{L}_{DSNT} + \mathscr{L}_V + \mathscr{L}_D$. Given binary visibility v^{gt} for each joint, regression loss $\mathscr{L}_{DSNT}$ and depth loss $\mathscr{L}_D$ are applied for $v^{gt} = 1$, and invisibility loss $\mathscr{L}_V$ is applied for $v^{gt} = 0$.

The *regression loss* $\mathscr{L}_{DSNT}$ is applied for the confidence maps V and coordinates $\mathbf{p}$ with the ground truth positions $\mathbf{p}^{gt}$ and binary visibility v^{gt} as:

$$\mathscr{L}_{DSNT} = \sum_{i=1}^{K} v_i^{gt} \cdot [||\mathbf{p}_i^{gt} - \mathbf{p}_i||_2^2 + \mathcal{D}(V_i || \mathcal{N}(\mathbf{p}_i^{gt}, \sigma I_2))] \quad (5)$$

$\mathcal{N}(\mu, \sigma)$ is a 2D Gaussian map drawn at μ with standard deviation σ ($\sigma = 1$ for training). $\mathcal{D}(\cdot||\cdot)$ is the Jensen-Shannon divergence to encourage H to resemble the 2D Gaussian map [25].

The *invisibility loss* $\mathscr{L}_V$ suppresses H to a zero heatmap for invisible joints:

$$\mathscr{L}_V = \sum_{i=1}^{K} (1 - v_i^{gt}) \cdot ||H_i||_2^2 \quad (6)$$

The invisibility loss forces the uniform distribution in V, which encourages the confidence value to be smaller for invisible joints.

The *depth loss* $\mathscr{L}_D$ is applied for depthmaps D with ground truth depthmaps D^{gt} and joint masks $\mathcal{M}(\mathbf{p}^{gt})$ as:

$$\mathscr{L}_D = \sum_{i=1}^{K} v_i^{gt} \cdot ||\mathcal{M}(\mathbf{p}_i^{gt}, \sigma I_2) \oslash (D_i - D_i^{gt})||_2^2 \quad (7)$$

$\mathcal{M}(\mu, \sigma)$ is a 2D binary maskmap drawn at μ with radius σ (set to 1.8 during training), and $\oslash$ is the Hadamard product. Note that the depth map is trained only for the interest joint area so that the outside area is left unchanged to prevent over-fitting which results in zero depthmap output when not using the maskmap [22].

The network is trained in multiple stages. First, the 2D layers are trained on the MPII Human Pose dataset [3] to learn low-level texture features. Only the regression loss $\mathscr{L}_{DSNT}$ is used in the training, while visibility is ignored. Then, the network is trained on our dataset in Sect. 3.2 with the full loss function $\mathscr{L}_{joint\_net}$. Intermediate supervision is applied during training.

We take advantage of the symmetry between the two downward-looking camera views to flip the right-sided image and use the same network as the left image. The output joint coordinates from the right image are then flipped back. This strategy allows a single network to be used at training and runtime for both views.

4.3 Temporally, Multi-view Consistent Joint Estimation

3D joints are detected in the left and right downward camera views independently and are reprojected into a single 3D space using the camera calibration matrices. Joints that are not consistent with their counterparts due to erroneous detection are filtered out such that the results are both multi-view-consistent and temporally coherent.

First, the raw detection of a joint is filtered out if its bone direction d_i is temporally inconsistent, which we define as a change of more than 30° between frames.

Next, the filtered measurements are used to estimate the multi-view-consistent and temporally-coherent joint position $X \in \mathbb{R}^3$, by minimizing the weighted sum $E_{proj} + w_d E_{dep} + w_l E_{len} + w_t E_{temp}$, where w_d, w_l, and w_t are non-negative weights. For torso joints including neck, hips, and shoulders, $w_d = 1, w_l = 0, w_t = 10$. $w_d = 2, w_l = 2, w_t = 1$ for arm joints, and $w_d = 1, w_l = 5, w_t = 2$ for leg joints.

The *projection cost* E_{proj} is defined as $\sum_{c=1}^{C} ||\mathbf{p}_c - P_c \cdot X||_2$, where C is the number of views, $\mathbf{p}_c$ is the 2D location measurement in camera image c, and P_c is camera c's projection matrix.

The *depth cost* E_{dep} is defined as $\sum_{c=1}^{C} ||d_c - T_c^{[3,:]} \cdot X||_2$, where d_c is the depth measurement in camera c, and $T_c^{[3,:]}$ is the third row of the extrinsic matrix of camera c.

Bone lengths are maintained over time, starting with the initialization and averaging with new detection measurements. The initial bone lengths are taken from the body model in its rest pose and scaled by the ratio between the model and detected spine lengths. The *bone length consistency* E_{len} is measured as $||X_l - ||X_p - X||_2||_2$, where X_p is the parent joint's position and X_l is the bone length of joint X.

The *temporal smoothness cost* E_{temp} is defined as $||X_{t-1} - X||_2$, where X_{t-1} is the joint position in the previous frame.

The estimated 3D joint positions X in headset space are transformed into joint positions J in 3D world space using the current estimated headset pose acquired via *VSLAM* [30] running in a separate thread at 35 fps.

The entire process, shown in Fig. 7, results in better accuracy than when using direct triangulation, even when joints are not detected in both views.

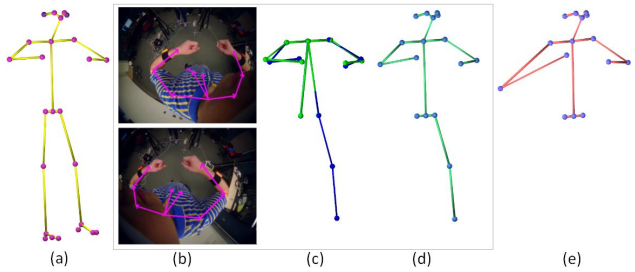

Figure 7: Consistent 3D joints. (a) Reference 3D joints. (b) Joint detections from left camera (top), and right camera (bottom). (c) 3D joints from left camera (blue), and right camera (green). (d) Joints reconstructed by our method. (e) Joints reconstructed using direct triangulation, for comparison.

4.4 Visual-Inertial Alignment

Human pose can be estimated with body-worn inertial sensors by using the sensor measurements to track the orientations of the corresponding bones. IMUs are typically calibrated using a specific initial pose [14, 39–41]. Prior methods assume that the sensors are placed accurately at designated poses (positions and orientations), and that the user assumes the correct body pose in the beginning. Even slightly misaligned body-worn IMUs can interfere with visual-inertial consistent pose estimation, yielding inaccurate results.

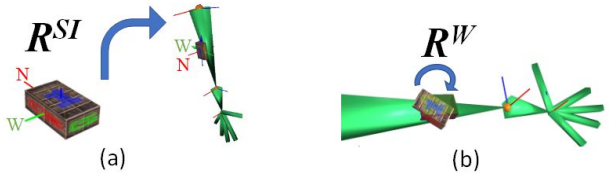

Figure 8: Coordinate frame transformations. (a) Rotation of inertial sensor to skeleton space R^{SI}, indicating the predefined wear pose. (b) IMU rotation offset R^W, used to compensate for misaligned IMUs.

We correct these inaccuracies by estimating an *IMU rotation offset* $R^W \in \mathbb{R}^{3\times3}$ using collected samples of visual and inertial pairs lower bone directions over time. It represents how much a sensor is offset from the assumed initial orientation of the bone (Fig. 8b).

The bone rotation R_t^S at time step t from Equation 1 can be computed for the lower bones from the IMUs mounted on them as:

$$R_t^S = R^W \cdot R_t^I \cdot (R^{SI})^{-1} \cdot R_0^S \tag{8}$$

R_t^I is the orientation read from the *I*nertial sensor at time t, R_0^S is the rotation from T_0^S in Equation 3, and $(R^{SI})^{-1}$ maps the coordinate frame $\mathcal{F}^S \mapsto \mathcal{F}^I$ (Fig. 8a).

We define the *I*nertial lower bone direction d_t^I as:

$$d_t^I = R_t^I \cdot (R^{SI})^{-1} \cdot d(R_0^S) \tag{9}$$

$d(R_0^S)$ indicates the bone direction in the rest pose from Equation 2.

The IMU rotation offset R^W is updated whenever measurements from the visual detector of the same bone are available, so that all prior bone directions $d(R_1^S), ..., d(R_t^S)$ agrees with the corresponding visual bone directions $d_1^V, ..., d_t^V$ from Equation 2. Note that $R^W = I_3$ when the sensor is worn in exactly the designated position and orientation. R^W can be estimated from a sequence of *V*isual d^V and *I*nertial d^I directions by solving the least square problem:

$$\min_{R^W} \sum_t ||d_t^V - R^W \cdot d_t^I||_2^2 \tag{10}$$

Solving Equation 10 for all available (d^I, d^V) pairs is computationally intensive. Instead, we group the visual-inertial pairs and update R^W using the online k-means algorithm described in algorithm 1 with a online k-d tree structure.

Algorithm 1: Online IMU Rotation Offset Calibration

Input: *I*nertial direction d^I, *V*isual direction d^V
Data: k clusters $\mathbf{c}$ in k-d Tree T, cluster $c_{min} \in \mathbf{c}$ with minimum nearest neighbor distance (nndist)
Output: IMU rotation offset R^W
$x \leftarrow$ next sample (d^I, d^V);
$c \leftarrow$ nearest(x) in T;
if $dist(x,c) < nndist(c_{min})$ **then**
  $c' \leftarrow$ average(x,c);
  replace c with c' in T;
else
  remove c_{min} from T;
  push x to T;
  find the new c_{min} in T;
end
Update R^W from $\mathbf{c}$ pairs using Equation 10

At runtime, we maintain a fixed $k = 200$ number of cluster pairs in the k-d tree. Our sampling strategy maximizes between-cluster distances, which favors uniform distribution of the clusters and minimizes the number of colinear samples.

The lower bone orientations R^S can always be estimated from R^W, regardless of their visibility, using Equation 8.

4.5 Temporal Visual-Inertial Orientation Network

Upper arm and thigh orientations can be estimated at every step using a sequence of forearm and lower leg motions, respectively, under the assumption that the movements of the lower and upper bones of the same limb are highly correlated [14, 41]. However, multiple upper arm or thigh orientations are possible for a single forearm or lower leg pose. To overcome this difficulty, our approach uses visual observations of the upper bones when available. In this section, we use the subscripts **i** and **u** to distinguish between the sensor-**i**nstrumented lower bones and the **u**ninstrumented upper bones.

The calibrated forearm and lower leg orientations $R_\mathbf{i}^S$ are computed using the IMU offset matrix $R_\mathbf{i}^W$ in Equation 8. Similarly, the raw accelerations $a_\mathbf{i}^I$ can be used to compute $a_\mathbf{i}^S = R_\mathbf{i}^H \cdot a_\mathbf{i}^I$ using the IMU acceleration offset matrix R^H, indicating the *H*eading reset, a rotation along the up direction computed from R^W.

We estimate the un-instrumented upper arm and thigh orientations $R_\mathbf{u}^S$ from a sequence of previous $R_\mathbf{i}^S$, $a_\mathbf{i}^S$ for the forearms and lower legs, as well as the availability of visual upper arm and thigh directions $d_\mathbf{u}^V$ from the visual detector in Sect. 4.3, while enforcing the constraint $d(R_\mathbf{u}^S) = d_\mathbf{u}^V$ from Equation 2. To be invariant to the body direction, $R_\mathbf{i}^S$, $a_\mathbf{i}^S$, and $d_\mathbf{u}^V$ are normalized with respect to the root joint (hip center) orientation R_{root}^S at time step t [14]:

$$R^N(t) = (R_{root}^S(t))^{-1} \cdot R_\mathbf{i}^S(t) \tag{11}$$

$a_\mathbf{i}^S \rightarrow a^N$, and $d_\mathbf{u}^V \rightarrow d^N$ are similarly normalized. We use $^\mathbf{N}$ to indicate the **N**ormalized torso space.

The input feature vector at time t is defined as:

$$x_t = [r_t, \omega_t, a_t, v_t \cdot d_t]^T \tag{12}$$

r_t denotes $[r_1^N(t), ..., r_4^N(t)]^T$ for 4 input bones. ω_t, a_t, and $v_t \cdot d_t$ are similarly defined. r_i^N is the vectorized R_i^N, and $\omega_i^N(t)$ is the angular velocity between $R_i^N(t)$ and $R_i^N(t-1)$. The input feature vector incorporates the lower bone motions represented by rotation, velocity, and acceleration. If the joints of the upper bone i are provided by the visual detector, its direction d_i^N is added and its visibility v_i is set to 1. Otherwise, experiments showed that using $v_i = 0.1^{-3}$ and $d_i^N = (1,1,1)$ yields better performance than setting both to 0. The dimension of x_t is $(9+3+3+3)\cdot 4 = 72$ for the 4 IMU-instrumented bones (r_t, ω_t, a_t) and for the 4 uninstrumented bones $(v_t \cdot d_t)$.

The output vector contains the vectorized uninstrumented bone orientations $y_t = [r_1^o(t), ..., r_4^o(t)]^T$. r_i^o are reshaped to the output orientations $R_i^o(t)$. The dimension of y_t is $(9)\cdot 4 = 36$ for the 4 upper arm and thigh bones.

Our network's task is to learn a function $f : \mathbf{x} \rightarrow y_t$ that predicts the uninstrumented bone orientations from a sequence of input features $\mathbf{x} = [x_{t-n+1}, ..., x_t]$. We employ a Transformer network, which has been shown to outperform LSTM in many applications [37]. The input sequence is composed of measurements from the last $n = 20$ frames [14]. The network architecture is shown in Fig. 9.

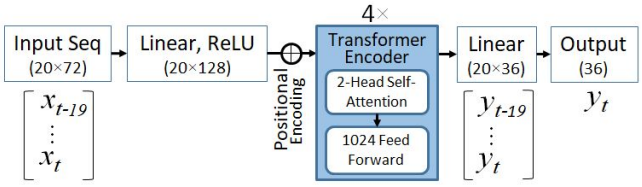

Figure 9: Temporal Visual-Inertial Orientation Network architecture. Using a sequence of visual-inertial input feature vectors $\mathbf{x}$, the uninstrumented orientations y are estimated. All layers use dropout 0.2 in training. The numbers in brackets indicate the output dimensions of each layer.

The network is trained with the following loss function:

$$\mathcal{L}_{bone\_net} = ||y - y^{gt}||_2^2 + \sum_{i=1}^{4} v_i^{gt} \cdot acos(d(R_i^o), d_i^{gt}) \tag{13}$$

The orientation loss is measured using the ground truth y^{gt}. $d(R^o)$ represents the output bone direction computed using Equation 2. It is penalized by the ground truth d^{gt} bone direction, which encourages the output bone direction to be consistent with the visual input bone direction if provided.This term is only computed if $v^{gt} = 1$.

At run-time, the estimated R^o in normalized torso space are transformed to $R_\mathbf{u}^S$ in world space using Equation 11.

4.6 Deformable Body Model Fitting

Our pipeline estimates the full body shape and pose: joint positions J and bone rotations R^S. Unobserved joint positions are recovered using forward kinematics from R^S and the corresponding bone lengths. The body shape is updated by solving Equation 4 using the full body joint positions J.

The bone rotations R^S are further corrected by using the detected visual direction outputs d^V when available. The estimated R^S are temporally coherent but the motion may be over-smoothed when sudden changes in motion or visibility occur along the edges of the camera images. This issue can be avoided by fitting bone orientations R^S closer to visual directions d^V, which encourages a quicker reaction to changes. The corrected bone rotations $\bar{R}^S$ can be estimated if d^V are available:

$$\bar{R}^S = R^{v2v}(d(R^S), \alpha \cdot d^V + (1-\alpha) \cdot d(R^S)) \cdot R^S \qquad (14)$$

$R^{v2v}(v_1, v_2)$ is the rotation from v_1 to v_2 vectors, and $\alpha = 0.8$ at run-time. The joint positions $\bar{J}$ are also updated by the forward kinematics using $\bar{R}^S$. The pose parameters T^M are estimated by using $\bar{R}^S$ and $\bar{J}$ in Equation 1 and Equation 3. The estimated joints $\bar{J}$ are transferred to the next frame for the temporally consistent joint estimation in Sect. 4.3.

5 Results and Evaluation

Our 3D pose estimation method is not directly comparable to any prior methods we are aware of. Outside-looking-in camera-based methods [13, 16, 21, 43] require all joints to be visible. Prior visual+inertial fusion approaches [18,34,39] additionally require more than 10 densely-worn IMUs. Our method uses as input stereo head-worn views that almost never capture the entire body, and only 4 inertial sensors worn on wrists and ankles. We compared our results with the following three baseline approaches:

HG3D (stereo stacked hourglass 3D) is a visual-only method that uses the 3D joint detector in Sect. 4.2 without the visibility awareness term in Equation 6 [22, 24, 25]. It detects both visible and invisible joints, and merges the joints from the two downward camera views as shown in Sect. 4.3 to produce full-body 3D joint positions. The 3D bone rotations are estimated from the detected joints using the inverse kinematics (IK) algorithm in [7]. We also separately evaluated a monocular stacked hourglass 3D on the publicly-available egocentric dataset in [45] and show competitive results in Table 3.

DIP is our implementation of Deep Inertial Poser [14], an IMU-based method which uses 6 sensors placed on wrists, ankles, torso and head. We used the ground truth values for head and torso orientations and accelerations, thus including only limb motions in the comparison. We also used ground truth body shapes and pre-calibrated inertial measurements. We included 20 past frames and 5 future frames, along with the best configuration of the LSTM architecture. In contrast, our own method estimates the body shapes and sensor calibrations at run-time, and does not use future frames.

Ours8 is a version of our method that uses 8 IMUs worn on wrists, ankles, upper arms and thighs. Since actual measurements are available, we skipped the temporal orientation network for upper arm and thigh bone estimation in Sect. 4.5. Instead, we applied the visual-inertial alignment in Sect. 4.4 to all 8 IMUs over time.

To assess the accuracy of our reconstruction results, we evaluated our system by comparing 3D joint position and orientation errors between our estimates and the ground truth. The results for the Ego-VIP dataset are shown in Table 2, broken down into three categories of joints: visible, occluded, and outside FoV. In all categories, our method significantly outperforms HG3D and DIP.

HG3D's accuracy is comparable with ours for visible joints, but its position errors are significantly higher for both occluded and outside-FoV joints. The orientations computed using IK are significantly less accurate than when acquired from inertial sensors. This comparison shows that even a few inertial sensors significantly improve pose accuracy in joint positions and orientations.

DIP shows significantly lower accuracy and higher variance than our method in both position and orientation. This comparison shows that incorporating even sparse visual information into an IMU-based method significantly stabilizes the temporal accuracy. For invisible joints, our method's accuracy drops significantly due to relying entirely on inertial sensors, while still outperforming DIP. Our Transformer network-based orientation estimation shows less variance than DIP's LSTM-based network.

Table 4 shows the position accuracy for each joint. Leg joints show significantly lower position accuracy due to decreased visibility and increased depths. Table 5 shows the orientation accuracy for each bone. Upper bones have lower orientation accuracy than lower bones because they are not instrumented with IMUs.

Fig. 10 shows a qualitative comparison. HG3D failed to detect correctly the occluding right knee and ankle in (a), and was unable to detect outside joints in (b). DIP underestimated the left knee lift in (a) and hand raise in (b), respectively. In (c), DIP outputs the wrong lower body pose due to the pose ambiguity from sparse IMU input. Our method shows significantly better pose estimates than HG3D and DIP in all cases.

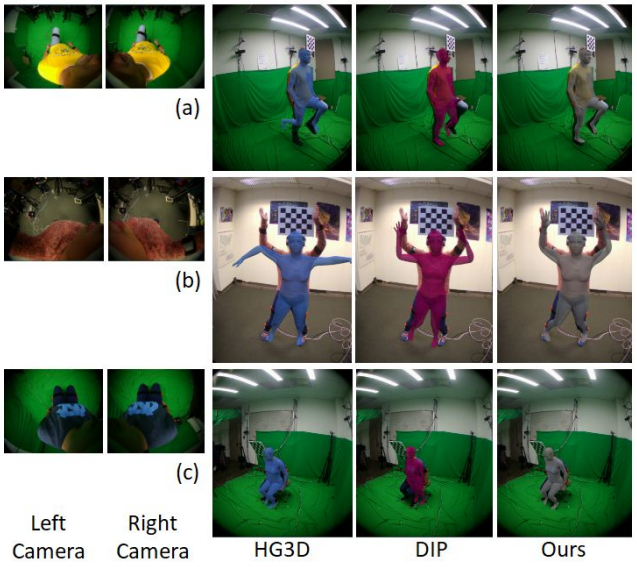

Figure 10: Qualitative evaluation. Selected frames, Ego-VIP dataset.

The Ours8 (dense-IMUs) variant of our method shows the best performance in all three categories because all bones are instrumented with IMUs. However, the accuracy of our proposed method, with only 4 IMUs, is comparable to that of Ours8, and both perform significantly better than either HG3D or DIP.

6 Applications

To showcase the real-time capability of our system, we demonstrate a remote Physical Therapy (PT) scenario in Virtual Reality (VR). The user wearing our prototype system and a trainer wearing an Oculus Quest VR headset are in different physical locations. Our learning-based pipeline estimates the current body configuration (10 body shape parameters and 24×3 pose parameters), which is sent to the trainer's VR headset over a wireless network via UDP. The VR headset uses the Unity Game Engine [35] to render the user's pre-scanned environment and body model from the trainer's viewpoint in real time. The trainer evaluates the user's PT motions and gives real time audio feedback on how to improve them. The trainer is provided with controller-based and physical locomotion to move around the user's environment. This demonstration shows that our system is able to reconstruct challenging and fast PT motions in real time and could be a viable tool for remote PT in the future. Fig. 11 shows an overview of this (unidirectional) PT demo system, and Fig. 1 (left 2×2 image group) shows sample results.

We also demonstrate our system outdoors, as shown in Fig. 1 (center 2×2 image group), using a backpack PC. The motion data was recorded and processed in real-time. Wearing the backpack, the

Table 2: Quantitative evaluation on the Ego-VIP dataset showing average joint position errors (cm) and orientation errors (degrees). The joint poses were evaluated for visible, occluded, and outside-camera-FoV cases. Methods: HG3D = Stereo Hourglass 3D (2 views); DIP (6 IMUs); Ours (2 views, 4 IMUs); Ours8 (2 views, 8 IMUs). The worst results are shown bolded.

	μ_{cm}^{tot}	σ_{cm}^{tot}	μ_{cm}^{vis}	σ_{cm}^{vis}	μ_{cm}^{occ}	σ_{cm}^{occ}	μ_{cm}^{out}	σ_{cm}^{out}	μ_{deg}^{tot}	σ_{deg}^{tot}	μ_{deg}^{vis}	σ_{deg}^{vis}	μ_{deg}^{occ}	σ_{deg}^{occ}	μ_{deg}^{out}	σ_{deg}^{out}
HG3D	3.69	4.44	2.67	2.81	6.18	5.58	**18.34**	**11.51**	**19.65**	**16.36**	**21.86**	**16.47**	**16.04**	**12.38**	**83.94**	**19.54**
DIP [14]	**6.06**	**5.32**	**4.33**	**4.31**	**10.52**	**6.91**	13.66	4.95	18.14	11.70	20.05	12.57	15.60	9.93	30.93	11.79
Ours	3.33	2.49	2.46	1.78	5.60	3.47	5.50	2.96	11.28	6.87	10.88	7.00	11.71	6.28	15.42	7.01
Ours8	3.17	1.68	2.44	1.31	5.08	2.16	4.50	1.63	8.76	4.72	7.74	4.33	9.99	4.99	11.78	4.29

Table 3: Performance of monocular HG3D on the Mo2Cap2 dataset [45] showing mean joint position errors (cm).

	Indoor (cm)	Outdoor (cm)
3DV'17 [20]	7.628	9.446
VNect [22]	9.785	11.375
Mo2Cap2 [45]	6.140	8.064
xR-EgoPose [33]	4.816	6.019
HG3D	8.680	8.823

Table 4: Per-joint average position errors (cm) for our method on Ego-VIP dataset. The joint poses were evaluated in visible, occluded, and outside-camera-FoV cases. The worst results are shown bolded.

	μ_{cm}^{tot}	σ_{cm}^{tot}	μ_{cm}^{vis}	σ_{cm}^{vis}	μ_{cm}^{occ}	σ_{cm}^{occ}	μ_{cm}^{out}	σ_{cm}^{out}
Neck	1.29	0.69	1.29	0.69	N/A	N/A	N/A	N/A
Shoulder	1.53	0.84	1.53	0.84	N/A	N/A	N/A	N/A
Hip	2.40	1.37	2.40	1.37	N/A	N/A	N/A	N/A
Elbow	2.34	1.76	2.15	1.28	3.55	2.60	**7.08**	**3.63**
Wrist	3.02	2.37	2.74	1.53	4.49	**4.30**	4.95	2.68
Knee	5.40	**3.84**	5.56	**4.42**	5.32	3.25	N/A	N/A
Ankle	**6.32**	3.73	**6.53**	3.78	**6.28**	3.60	N/A	N/A

Table 5: Per-bone average orientation errors (degrees) for our method on the Ego-VIP dataset, using only forearm- and lower-leg IMUs; upper bones estimated. The worst results are shown bolded.

	μ_{deg}^{tot}	σ_{deg}^{tot}	μ_{deg}^{vis}	σ_{deg}^{vis}	μ_{deg}^{occ}	σ_{deg}^{occ}	μ_{deg}^{out}	σ_{deg}^{out}
Up Arm	**12.7**	**8.5**	12.5	**8.2**	**13.1**	**8.2**	**25.9**	**9.7**
Thigh	12.4	7.1	**15.0**	7.9	11.1	6.2	N/A	N/A
Forearm	7.4	5.0	7.2	4.7	7.8	5.6	11.7	5.7
Lo Leg	12.5	6.3	12.2	7.3	12.5	6.0	N/A	N/A

user performed a number of standard soccer exercises. Our method successfully reconstructed the movements in a grassy area of about 50 square meters. This showcases the mobility of our system.

In both demos, the user's environments were pre-reconstructed using Agisoft's Metashape software [1]. The body texture was derived from two full-body images of the user (front and back). We used *SMPLify-X* [27] to fit the SMPL body model to the body and facial keypoints [6] acquired from the images. The colors from the images were then rasterized to a canonical UV map based on the established correspondence between the fitted meshes and the body part segmentations [11].

Our prototype system runs at 37 fps on a desktop PC (Intel Xeon Gold 6242, 2.8GHz, 128 GB RAM, with NVIDIA Quadro RTX 6000) and at 30 fps on a backpack PC (Intel i7-8850H, 2.6GHz, 32GB RAM with NVIDIA GeForce RTX 2080).

7 Future Work

Our current system's limitations offer opportunities for future work. Since our system only tracks the user's limbs, it does not model interactions with the the environment, nor is it able to detect topological or texture changes in the surface of the body model. We plan to add support for interactions with objects such as moving a chair, topological changes such as putting on a tie, and texture changes such as wearing a different shirt. We also plan to extend our approach to be physically plausible by estimating 3D environment contacts, as well as to more realistic shapes with other body models [26, 27].

The joint position accuracy is highly dependent on the *VSLAM* result, which is used to transform the estimated joints into world space. If *VSLAM* is unstable or inaccurate over time, the body pose accuracy drops as well. In the next iteration of our system, we plan to use multiple forward cameras and integrate an IMU into the headset for more robustness in the head pose estimation.

In our current pipeline, the results of the 3D joint detection network are fed into the temporal orientation network. If the 3D joints are detected erroneously, such errors are propagated throughout. We plan to investigate a combined network, as well as improving robustness against erroneous detections.

Finally, unlike our current PT prototype (Fig. 11), future application prototypes will demonstrate bi-directional telepresence.

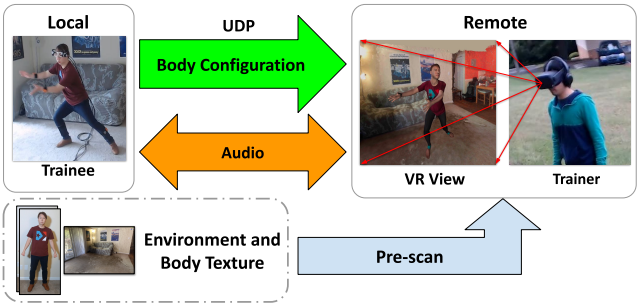

Figure 11: Interactive Physical Therapy application in VR. The real-time body reconstruction is only transmitted from trainee to trainer. The trainer's VR display shows the trainee's full-body performance using the pre-scanned environment and body texture. The trainer provides real-time feedback via audio.

8 Conclusion

We presented a real-time egocentric 3D capture system as a step toward a fully mobile telepresence system. Our system makes use of visual and inertial sensors that are either easy to embed into or are already present in commonly worn personal accessories: eyeglasses, wristwatches, and shoes.

The eyeglasses form factor makes visibility challenging, while the small number of inertial sensors makes the full body pose difficult to estimate. To address these challenges, our system combines visual and inertial information and shows improved full-body pose estimation compared to visual-only or inertial-only information.

In the future, as cameras and IMUs become smaller and more ubiquitous, we anticipate non-encumbering and easy-to-use real-time successors to our mobile telepresence prototype to become commonplace and useful for many everyday communication tasks.

Acknowledgments

Jim Mahaney helped extensively with hardware prototypes and experimental capture. We thank our collaborators from Ximmerse [44] for the headset design in Figure 2d. This work was partially supported by National Science Foundation Awards 1405847, 1718313, 1840131, and by the BeingTogether Centre, a collaboration between Nanyang Technological University (NTU) Singapore and University of North Carolina (UNC) at Chapel Hill, supported by UNC and the Singapore National Research Foundation, Prime Minister's Office, Singapore under its International Research Centres in Singapore Funding Initiative.

REFERENCES

[1] Agisoft Metashape Standard. https://www.agisoft.com/downloads/installer/, 2020.

[2] R. Alp Güler, N. Neverova, and I. Kokkinos. Densepose: Dense human pose estimation in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7297–7306, 2018.

[3] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[4] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black. Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In *European Conference on Computer Vision*, pp. 561–578. Springer, 2016.

[5] J. Butepage, M. J. Black, D. Kragic, and H. Kjellstrom. Deep representation learning for human motion prediction and classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6158–6166, 2017.

[6] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh. Openpose: realtime multi-person 2d pose estimation using part affinity fields. *arXiv preprint arXiv:1812.08008*, 2018.

[7] Y.-W. Cha, T. Price, Z. Wei, X. Lu, N. Rewkowski, R. Chabra, Z. Qin, H. Kim, Z. Su, Y. Liu, et al. Towards fully mobile 3d face, body, and environment capture using only head-worn cameras. *IEEE transactions on visualization and computer graphics (TVCG), Proceedings of ISMAR 2018, October, Munich, Germany*, 24(11):2993–3004, 2018.

[8] L. Chan, C.-H. Hsieh, Y.-L. Chen, S. Yang, D.-Y. Huang, R.-H. Liang, and B.-Y. Chen. *Cyclops: Wearable and Single-Piece Full-Body Gesture Input Devices*, p. 3001–3009. Association for Computing Machinery, New York, NY, USA, 2015.

[9] Y. Cheng, B. Yang, B. Wang, W. Yan, and R. T. Tan. Occlusion-aware networks for 3d human pose estimation in video. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 723–732, 2019.

[10] R. Dabral, A. Mundhada, U. Kusupati, S. Afaque, A. Sharma, and A. Jain. Learning 3d human pose from structure and motion. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 668–683, 2018.

[11] K. Gong, X. Liang, Y. Li, Y. Chen, M. Yang, and L. Lin. Instance-level human parsing via part grouping network. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 770–785, 2018.

[12] M. Habermann, W. Xu, M. Zollhoefer, G. Pons-Moll, and C. Theobalt. Livecap: Real-time human performance capture from monocular video. *ACM Transactions on Graphics (TOG)*, 38(2):1–17, 2019.

[13] M. Habermann, W. Xu, M. Zollhofer, G. Pons-Moll, and C. Theobalt. Deepcap: Monocular human performance capture using weak supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5052–5063, 2020.

[14] Y. Huang, M. Kaufmann, E. Aksan, M. J. Black, O. Hilliges, and G. Pons-Moll. Deep inertial poser: Learning to reconstruct human pose from sparse inertial measurements in real time. *ACM Transactions on Graphics (TOG), Proceedings of SIGGRAPH Asia 2018*, 37:185:1–185:15, Nov. 2018.

[15] H. Jiang and K. Grauman. Seeing invisible poses: Estimating 3d body pose from egocentric video. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3501–3509. IEEE, 2017.

[16] M. Kocabas, N. Athanasiou, and M. J. Black. Vibe: Video inference for human body pose and shape estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5253–5263, 2020.

[17] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black. Smpl: A skinned multi-person linear model. *ACM transactions on graphics (TOG), Proceedings of SIGGRAPH Asia 2015*, 34(6):248, 2015.

[18] C. Malleson, A. Gilbert, M. Trumble, J. Collomosse, A. Hilton, and M. Volino. Real-time full-body motion capture from video and imus. In *2017 International Conference on 3D Vision (3DV)*, pp. 449–457. IEEE, 2017.

[19] J. Martinez, M. J. Black, and J. Romero. On human motion prediction using recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2891–2900, 2017.

[20] D. Mehta, H. Rhodin, D. Casas, P. Fua, O. Sotnychenko, W. Xu, and C. Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *2017 international conference on 3D vision (3DV)*, pp. 506–516. IEEE, 2017.

[21] D. Mehta, O. Sotnychenko, F. Mueller, W. Xu, M. Elgharib, P. Fua, H.-P. Seidel, H. Rhodin, G. Pons-Moll, and C. Theobalt. Xnect: Real-time multi-person 3d human pose estimation with a single rgb camera. *arXiv preprint arXiv:1907.00837*, 2019.

[22] D. Mehta, S. Sridhar, O. Sotnychenko, H. Rhodin, M. Shafiei, H.-P. Seidel, W. Xu, D. Casas, and C. Theobalt. Vnect: Real-time 3d human pose estimation with a single rgb camera. *ACM Transactions on Graphics (TOG), Proceedings of SIGGRAPH 2017*, 36(4):44, 2017.

[23] Microsoft Hololens. https://developer.microsoft.com/en-us/windows/mixed-reality/hololens_hardware_details, 2016. Accessed: 2017-03-15.

[24] A. Newell, K. Yang, and J. Deng. Stacked hourglass networks for human pose estimation. In *European conference on computer vision*, pp. 483–499. Springer, 2016.

[25] A. Nibali, Z. He, S. Morgan, and L. Prendergast. Numerical coordinate regression with convolutional neural networks. *arXiv preprint arXiv:1801.07372*, 2018.

[26] A. A. A. Osman, T. Bolkart, and M. J. Black. Star: Sparse trained articulated human body regressor. In *European Conference on Computer Vision (ECCV)*, vol. LNCS 12355, pp. 598–613, Aug. 2020.

[27] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. A. Osman, D. Tzionas, and M. 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)*, 2019.

[28] H. Rhodin, C. Richardt, D. Casas, E. Insafutdinov, M. Shafiei, H.-P. Seidel, B. Schiele, and C. Theobalt. Egocap: egocentric marker-less motion capture with two fisheye cameras. *ACM Transactions on Graphics (TOG), Proceedings of SIGGRAPH Asia 2016*, 35(6):162, 2016.

[29] T. Shiratori, H. S. Park, L. Sigal, Y. Sheikh, and J. K. Hodgins. Motion capture from body-mounted cameras. *ACM Transactions on Graphics (TOG)*, 30(4):31, 2011.

[30] S. Sumikura, M. Shibuya, and K. Sakurada. Openvslam: a versatile visual slam framework. In *Proceedings of the 27th ACM International Conference on Multimedia*, pp. 2292–2295, 2019.

[31] X. Sun, B. Xiao, F. Wei, S. Liang, and Y. Wei. Integral human pose regression. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 529–545, 2018.

[32] J. Tautges, A. Zinke, B. Krüger, J. Baumann, A. Weber, T. Helten, M. Müller, H.-P. Seidel, and B. Eberhardt. Motion reconstruction using sparse accelerometer data. *ACM Transactions on Graphics (ToG), Proceedings of SIGGRAPH 2011*, 30(3):18, 2011.

[33] D. Tome, P. Peluse, L. Agapito, and H. Badino. xr-egopose: Egocentric 3d human pose from an hmd camera. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 7728–7738, 2019.

[34] M. Trumble, A. Gilbert, C. Malleson, A. Hilton, and J. P. Collomosse. Total capture: 3d human pose estimation fusing video and inertial sensors. In *Proceedings of the British Machine Vision Conference (BMVC)*, pp. 14.1–14.13, September 2017. doi: 10.5244/C.31.14

[35] Unity Game Engine. https://unity.com/, 2020.

[36] G. Varol, D. Ceylan, B. Russell, J. Yang, E. Yumer, I. Laptev, and C. Schmid. Bodynet: Volumetric inference of 3d human body shapes. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 20–36, 2018.

[37] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

[38] R. Villegas, J. Yang, Y. Zou, S. Sohn, X. Lin, and H. Lee. Learning to generate long-term future via hierarchical prediction. In *Proceedings*

of the 34th International Conference on Machine Learning (ICML), pp. 3560–3569. JMLR. org, 2017.

[39] T. von Marcard, R. Henschel, M. J. Black, B. Rosenhahn, and G. Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 601–617, 2018.

[40] T. Von Marcard, G. Pons-Moll, and B. Rosenhahn. Human pose estimation from video and imus. *IEEE transactions on pattern analysis and machine intelligence (TPAMI)*, 38(8):1533–1547, 2016.

[41] T. von Marcard, B. Rosenhahn, M. Black, and G. Pons-Moll. Sparse inertial poser: Automatic 3d human pose estimation from sparse imus. *Computer Graphics Forum 36(2), Proceedings of the 38th Annual Conference of the European Association for Computer Graphics (Eurographics)*, pp. 349–360, 2017.

[42] C. Wang, J. Miguel Buenaposada, R. Zhu, and S. Lucey. Learning depth from monocular videos using direct methods. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2022–2030, 2018.

[43] D. Xiang, H. Joo, and Y. Sheikh. Monocular total capture: Posing face, body, and hands in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 10965–10974, 2019.

[44] Ximmerse. `https://www.ximmerse.com/en/`.

[45] W. Xu, A. Chatterjee, M. Zollhofer, H. Rhodin, P. Fua, H.-P. Seidel, and C. Theobalt. Mo2cap2: Real-time mobile 3d motion capture with a cap-mounted fisheye camera. *IEEE transactions on visualization and computer graphics (TVCG), Proceedings of IEEE VR*, 2019.

[46] T. Yu, Z. Zheng, Y. Zhong, J. Zhao, Q. Dai, G. Pons-Moll, and Y. Liu. Simulcap: Single-view human performance capture with cloth simulation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5499–5509. IEEE, 2019.

[47] F. Zhang, X. Zhu, and M. Ye. Fast human pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3517–3526, 2019.