International Journal of Computer Vision
https://doi.org/10.1007/s11263-019-01270-5

Real-Time Multi-person Motion Capture from Multi-view Video and IMUs

Charles Malleson[1] · **John Collomosse**[1] · **Adrian Hilton**[1]

Received: 4 January 2019 / Accepted: 25 November 2019

Abstract

A real-time motion capture system is presented which uses input from multiple standard video cameras and inertial measurement units (IMUs). The system is able to track multiple people simultaneously and requires no optical markers, specialized infra-red cameras or foreground/background segmentation, making it applicable to general indoor and outdoor scenarios with dynamic backgrounds and lighting. To overcome limitations of prior video or IMU-only approaches, we propose to use flexible combinations of multiple-view, calibrated video and IMU input along with a pose prior in an online optimization-based framework, which allows the full 6-DoF motion to be recovered including axial rotation of limbs and drift-free global position. A method for sorting and assigning raw input 2D keypoint detections into corresponding subjects is presented which facilitates multi-person tracking and rejection of any bystanders in the scene. The approach is evaluated on data from several indoor and outdoor capture environments with one or more subjects and the trade-off between input sparsity and tracking performance is discussed. State-of-the-art pose estimation performance is obtained on the Total Capture (mutli-view video and IMU) and Human 3.6M (multi-view video) datasets. Finally, a live demonstrator for the approach is presented showing real-time capture, solving and character animation using a light-weight, commodity hardware setup.

Keywords Pose estimation · Motion capture · IMU · Multi-view video · Real-time · Multi-person

1 Introduction

Real-time capture of human motion is of considerable interest in various domains including entertainment and the life sciences. Recent advances in computer vision (e.g. Captury 2017) and the availability of commodity wireless inertial sensors (e.g. Roetenberg et al. 2013; PerceptionNeuron 2017) are beginning to take motion capture from constrained studio settings to more natural, outdoor environments, and with less encumbrance of the performers from specialized costumes and optical marker setups traditionally required (e.g. Vicon 2017; OptiTrack 2017), while still retaining a high level of capture fidelity.

In the proposed approach, we fuse multi-modal input from inertial sensors and multiple cameras to produce an estimate of the full 3D pose of one or more subjects in real time without requiring optical markers or a complex hardware setup (Fig. 1). A solver optimizes the kinematic pose of the subject based on a cost function comprising orientation, acceleration, 2D position and statistical pose prior terms. The orientation and acceleration constraints are provided by a sparse set of inertial measurement units (IMUs) attached to body segments, and positional constraints are obtained from 2D joint detections from video cameras (Cao et al. 2017). Combining video and IMU data improves the tracking performance compared to one or the other. The IMUs provide full rotational information for body segments, while the video information provides drift-free 3D global position information.

This work is an extension of the work of Malleson et al. (2017). The proposed approach leads to significant improve-

Communicated by Edmond Boyer.

Electronic supplementary material The online version of this article (https://doi.org/10.1007/s11263-019-01270-5) contains supplementary material, which is available to authorized users.

✉ Charles Malleson
c.malleson@surrey.ac.uk

John Collomosse
j.collomosse@surrey.ac.uk

Adrian Hilton
a.hilton@surrey.ac.uk

[1] Centre for Vision, Speech and Signal Processing (CVSSP), University of Surrey, Guildford GU2 7XH, UK

Published online: 17 December 2019

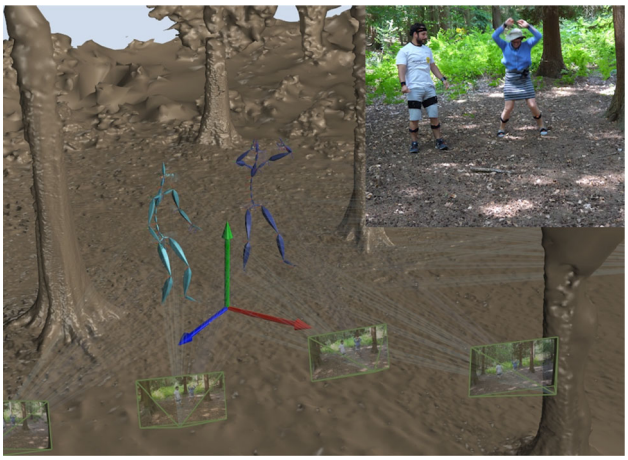


Fig. 1 Real-time full-3D motion capture of multiple people in unconstrained environments without visual markers

ments in solved pose accuracy compared to Malleson et al. (2017) and supports tracking of multiple subjects. Furthermore, several additional experiments have been performed, with additional test datasets and comparisons to further methods from the literature. The contributions of this work are as follows:

(1) Improved kinematic pose accuracy through, (a) use of calibrated 3D offsets between detected keypoint locations and solve skeleton, (b) use of an expanded 25 keypoint CPM detection input (vs. 12 used in Malleson et al. (2017)), and (c) addition of a floor penetration term. (2) Simultaneous tracking of multiple subjects through use of an efficient detection sorting mechanism which can operate in the presence of bystanders and mis-detections in the images. (3) Extensive further evaluation with additional test datasets, more detailed error statistics and treatment of single modality and monocular input cases. (4) Creation of a new multi-view video plus IMU dataset *Outdoor Duo* featuring two subjects performing diverse actions in uncontrolled outdoor settings. (5) Exposition of a light-weight demonstrator implementation with live motion output to a game engine environment and displayed in VR.

The remainder of the paper is structured as follows. Section 2 puts this work in the context of related work. Section 3 describes the approach. In Sect. 4, quantitative and qualitative evaluation is presented followed by a description of the implementation of our live demonstrator system. Finally, conclusions and future work are presented in Sect. 5.

2 Related Work

Various approaches exist in the literature for estimating human pose from video and other sensor data. To the best of our knowledge, the proposed approach is the first to use multiple views of natural video along with IMU input to estimate the global kinematic pose of one or more subjects in real time. Below we cover the most closely related prior works.

Convolutional Pose Machines (CPMs), proposed by Wei et al. (2016) and Cao et al. (2017), use deep neural networks to estimate 2D pose (joint and surface keypoint locations) for multiple people from a single image, with video rate detection possible using GPU acceleration. As the pose is 2D, there is no explicit conservation of bone lengths, and the axial rotations of limbs are not estimated. Our approach uses keypoint detections from CPMs over multiple camera views in combination with IMU data as input in a robust kinematic optimization to obtain accurate global 3D pose in real time. This kinematic pose output inherently maintains bone length and can be used directly to drive 3D character animation. Various works use neural networks to infer 3D pose directly from monocular cameras, e.g. Li et al. (2017), Tekin et al. (2016), Lin et al. (2017), Martinez et al. (2017), Mehta et al. (2018) and Tome et al. (2018). In Tome et al. (2017), CPMs are extended to 'lift' 3D pose from a single RGB image by incorporating knowledge of plausible human poses in the training, while in Tome et al. (2018) the lifting approach is improved by refining detection estimates in an end-to-end approach. In 'VNect', proposed by Mehta et al. (2017), 3D pose is estimated in real time from a single camera using convolutional neural networks (CNNs) and kinematic fitting, while Zhou et al. (2016) perform 2D joint detection using CNNs and estimate 3D pose using offline Expectation-Maximization over an entire sequence. Mehta et al. (2018) perform multi-person 3D pose estimation from monocular video using CNNs along with a occlusion-robust pose-map (ORPM) formulation for handling of strong partial occlusion. In Zanfir et al. (2018), the 3D pose and shape of multiple people are estimated from monocular video by fitting parametric body models using optimization of an estimated ground plane, exclusion of multiple occupancy, and temporal trajectory optimization while in the DensePose approach of Alp Güler et al. (2018), surface meshes are fitted to single view images producing dense pose correspondences. In unconstrained settings, monocular approaches are inherently subject to ambiguity in depth limiting their ability to recover absolute position and scale.

Elhayek et al. (2015) use CNNs to track multiple subjects from a sparse set of two or more cameras (the approach is not real-time on current hardware, with a reported runtime of > 1 second per frame for a single subject using three cameras). Rhodin et al. (201b) use 2D joint keypoint detections as well as body contours to jointly optimize body shape and pose from a small number of camera views while Joo et al. (2018) capture body, hand and facial motion from multiple view video by offline optimization of a deformable 3D body model.

IMUs and multi-view video data have been combined by Von Marcard et al. (2016) to exploit the complementary prop-

erties of the data sources, i.e. drift free position from video and 3D limb orientation from IMUs. However no comparison is performed against commercial reference-quality motion capture (instead the results are compared with respect to consistency with silhouettes and IMU measurements), and processing time is not specified. Andrews et al. (2016) perform real-time body tracking using a sparse set of labelled optical markers, IMUs, and a motion prior in an inverse dynamics formulation. In contrast, our approach does not use optical markers and does not require setting up a physics model of the subject. The 'Sparse Inertial Poser' (SIP) system proposed by von Marcard et al. (2017) uses orientation and acceleration from 6 IMUs as input and is assisted by a prior pose model in the form of the SMPL body model (Loper et al. 2015). However, because SIP processes sequences as a batch it is not suitable for online operation and the lack of visual information makes it susceptible to drift in global position. In 'Deep Inertial Poser' (DIP), Huang et al. (2018) recover local pose using a sparse set of 6 IMUs using a sliding window of frames for online operation. Our system uses input from cameras in addition to sparse IMUs, processes sequences online in real-time and is able to recover drift-free global position. In their 'Video Inertial Poser' (VIP) approach, von Marcard et al. (2018) estimate the 3D pose of multiple subjects given monocular camera and IMU input. Association between 2D detections and 3D subjects are obtained using a graph based optimization enforcing 3D to 2D coherency within a frame and across long range frames. Parameters for the shape and pose of the SMPL statistical body model (Loper et al. 2015) are jointly optimized along with IMU heading drift. In VIP, all frames are optimized simultaneously, precluding real-time operation. In contrast our approach, with its frame-to-frame processing and simple detection sorting mechanism operates online with multiple subjects at at video rates.

Trumble et al. (2016) use convolutional neural networks on multi-view video data to perform human pose estimation. In subsequent work, Trumble et al. (2017) combined video and IMU input in a deep learning framework, including using an LSTM (long short term memory, Hochreiter and Schmidhuber (1997)) for temporal prediction to reduce noise, and later using a deep autoencoder (Trumble et al. 2018) to recover a both skeletal pose (as joint positions) and detailed 3D body shape over time. These approaches require extensive training from multi-view video data and the axial rotation of the limbs cannot be recovered since the input is based on visual hulls. Furthermore, in common with all visual-hull based approaches, they require accurate foreground-background segmentation, typically requiring controlled capture conditions, extensive computation time or manually-assisted segmentation. In contrast, our method requires minimal, simple training of the pose prior, while using a pre-trained CPM detector for 2D detections. By incorporating IMU data, our method is able to recover axial rotation of the limbs while handling dynamic backgrounds and occlusions.

Introducing environmental constraints into video-based pose estimation can help resolve ambiguities in detection. For instance, Rosenhahn et al. (2008) integrate knowledge of the floor location to discourage vertices in a solved body surface mesh from penetrating the floor. In the proposed approach, which does not consider surface geometry, a simple floor penetration constraint is applied to specific targets in the skeleton (i.e. the foot and toe joints).

Various recent approaches to real-time body tracking use other types of capture hardware, for example two fisheye cameras attached to the subject (Rhodin et al. 2016a), Kinect RGBD cameras (Wei et al. 2012; Ichim and Tombari 2016), Kinect plus IMUs (Helten et al. 2013), HTC Vive infra-red VR controllers strapped to the limbs (IKinema 2017), or radio frequency (RF) equipment (Zhao et al. 2018). Such hardware is typically limited to indoor capture environments.

In Malleson et al. (2017), full-body markerless tracking of a single subject is performed in real-time in unconstrained environments using multiple-view video with as few as two cameras and 6 IMUs as input, recovering the full DoF including axial rotation and drift-free global position. In this work, we extend the approach of Malleson et al. to provide more accurate pose and to handle multiple people simultaneously. Furthermore, we demonstrate operation with further reduced camera/IMU input configurations and in less constrained capture environments.

3 Method

The proposed approach obtains the kinematic pose of multiple subjects in real time given input from any configuration of IMU or video input. First, the notation and pose parametrization are presented in Sect. 3.1 followed by descriptions of the kinematic pose cost function and its optimization in Sect. 3.2. Next, in Sect. 3.3, details of the generation and use of 2D keypoint detections are presented, i.e. efficiently performing 2D detection across multiple camera views, assigning 2D detected people to their corresponding tracked subjects, and use of optimized offsets between detections and their corresponding bones in the skeleton. Implementation details, including details of the capture setups used and live implementation of our approach are deferred to the results section.

3.1 Notation and Skeleton Parametrization

For each tracked subject, a kinematic skeleton is defined consisting of a pre-defined hierarchy of n_b rigid bones, b attached at joints. The root bone $b = 1$ (i.e. the hips) has a global position, $\mathbf{t}_1$ and orientation, $\mathbf{R}_1$. Each child bone, $b \in [2, n_b]$ is attached to its parent with a fixed translational offset, $\mathbf{t}_b$, and

pose-varying rotation, $\mathbf{R}_b$, w.r.t. the parent bone coordinates. In this work, $n_b = 21$ bones are used. The total degrees of freedom (DoF) are $d = 3 + 3 \times 21 = 66$, consisting of the root translation and 3 rotational degrees of freedom per joint. The skeleton topology is the same for all subjects and appropriate bone lengths are determined per subject (either from the calibrated skeleton obtained from the optical reference data, if available, or by scaling a reference skeleton according to the subject's known height).

We encode the pose of the skeleton as a single 66-dimensional vector $\boldsymbol{\theta}$ containing the 3D global translation of the root, followed by the stacked local joint rotations of each bone (including the root), represented as 3D angle-axis vectors (i.e. the axis of rotation multiplied by the angle of rotation in radians). This parameter vector is the variable which is optimized, with the root translation $\mathbf{t}_1$ and joint rotations $\mathbf{R}_b$ being extracted and used in calculations as applicable.

For each bone, b, the global rigid body transform $\mathbf{T}_b^g$ is computed by concatenating bone offset and joint rotation transforms along the kinematic chain as follows:

$$\mathbf{T}_b^g(\boldsymbol{\theta}) = \prod_{b' \in \mathcal{P}(b)} \left[\begin{array}{c|c} \mathbf{R}_{b'} & \mathbf{t}_{b'} \\ \hline 0 & 1 \end{array}\right] \tag{1}$$

where $\mathcal{P}(b)$ is the ordered set of parent joints of bone b.

We define a set of n_i IMU track targets, i, each attached to a bone b_i. The rotational and translational offsets of the IMU w.r.t. the bone are denoted $\mathbf{R}_{ib}$ and $\mathbf{t}_{ib}$, respectively. The rotational transform between each IMU reference frame and the global coordinates is denoted $\mathbf{R}_{ig}$. IMU orientation measurements (w.r.t. the IMU inertial reference frame) and acceleration measurements (w.r.t. the IMU device frame) are denoted $\mathbf{R}_i$ and $\mathbf{a}_i$, respectively. Likewise, we define a set of n_p positional track targets, p, each attached to a bone b_p with translational offset $\mathbf{t}_{pb}$ w.r.t. the bone. Note that here we use the term 'track target' to refer to any specific point on or in the body for which motion is estimated, not a physical optical marker. In our approach 2D joint positions are estimated using natural images and no visual markers are required.

Finally, we define a set of n_c cameras, c with calibrated 3×4 projection matrices $\mathbf{P}_c$ and let $\mathbf{t}_p^c$ denote the 2D position measurement for track target p in the local coordinates of camera c.

3.2 Pose Optimization

The kinematic pose of each tracked subject is optimized independently according to the following cost function:

$$E(\boldsymbol{\theta}) = E_{Data}(\boldsymbol{\theta}) + E_{Prior}(\boldsymbol{\theta}). \tag{2}$$

The data cost

$$E_{Data}(\boldsymbol{\theta}) = \overbrace{E_P(\boldsymbol{\theta})}^{Video} + \overbrace{E_R(\boldsymbol{\theta}) + E_A(\boldsymbol{\theta})}^{IMU} \tag{3}$$

incorporates positional constraints, E_P based on 2D keypoint detections from the input video as well as orientation and acceleration constraints, E_R and E_A from the input IMU measurements. The prior cost

$$E_{Prior}(\boldsymbol{\theta}) = \overbrace{E_{PP}(\boldsymbol{\theta}) + E_{PD}(\boldsymbol{\theta})}^{PCA} + \overbrace{E_{FP}(\boldsymbol{\theta})}^{Floor} \tag{4}$$

contain PCA pose prior projection and deviation terms, E_{PP} and E_{PD} as well as an optional floor penetration term, E_{FP}.

The proposed cost function extends that of Malleson et al. (2017) by including the floor penetration term. Each term in Eq. 2 is described in the following subsections, where solved values have a '^' circumflex and their dependence on $\boldsymbol{\theta}$ is omitted for clarity. Unless otherwise specified, values are for the current frame, t. In order to assess the effectiveness of each term in the cost function, an ablation study is presented in Sect. 4.1.2.

3.2.1 Position Term

For each 2D positional measurement from each camera, a constraint is added which seeks to minimize the Euclidean distance between the measured 2D location in camera coordinates and the solved global track target location projected into the camera (Fig. 2).

The solved global track target location, $\hat{\mathbf{t}}_p^g$ is determined by applying the translational offset $\mathbf{t}_{pb}$ to the global bone transform $\mathbf{T}_{b_p}^g$ (as calculated according to Eq. 1):

$$\hat{\mathbf{t}}_p^g = \boldsymbol{\tau}_t\left(\boldsymbol{\tau}_T(\mathbf{t}_{pb}) \cdot \mathbf{T}_{b_p}^g\right) \tag{5}$$

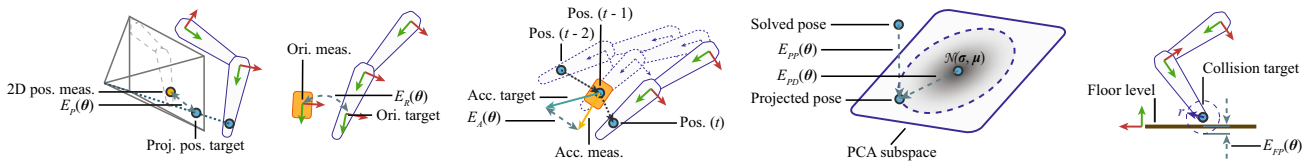
where the operators $\boldsymbol{\tau}_T(\cdot)$ and $\boldsymbol{\tau}_t(\cdot)$ are shorthand for creating a transform matrix from a translation vector and extracting the translation vector from a transform matrix, respectively. This global target position is projected into each camera to obtain 2D solved targets $\hat{\mathbf{t}}_p^c$ in camera coordinates:

$$\hat{\mathbf{t}}_p^c = \mathrm{dh}(\mathbf{P}_c \hat{\mathbf{t}}_p^g) \tag{6}$$

where the operator $\mathrm{dh}(\cdot)$ performs de-homogenization of a homogeneous vector.

The position cost is defined as

$$E_P(\boldsymbol{\theta}) = \sum_{c \in [1,n_c]} \sum_{p \in [1,n_p]} \rho_P\left(\lambda_P w_p^c \left\|\hat{\mathbf{t}}_p^c - \mathbf{t}_p^c\right\|_2^2\right) \tag{7}$$



(a) Position term **(b)** Orientation term **(c)** Acceleration term **(d)** PCA prior term **(e)** Floor penetration term

Fig. 2 The solver objective function (Eq. 2) comprises several terms, visualized in this figure. These are described in detail in Sect. 3.2 and their relative importance is assessed through an ablation study in Sect. 4.1.2

where $w_p^c \in [0, 1]$ is a confidence weighting for constraint p obtained from the image-based keypoint position measurement mechanism (in our case from a CPM-based detection, see Sect. 3.3), λ_P is a position constraint weighting factor, $\rho_P(\cdot)$ is a loss function (see Sect. 3.2.6). The confidence weighting w_p^c and loss function enable robust output pose estimates in spite of persistently high levels of noise and frequent outliers in input position detections.

Note that in Malleson et al. (2017), only *joint* keypoints were used as track targets and these were assumed to correspond on average, to the joints in the reference skeleton and thus have zero offset w.r.t. the bone ($\mathbf{t}_{pb} = \mathbf{0}$). In order to account for any systematic offset between the reference detected keypoints and the corresponding joint in the reference skeleton, and to allow *surface* keypoints such as the ears and eyes to be used as well, we propose to calibrate offsets $\mathbf{t}_{pb}$ for the reference skeleton and detector used (see Sect. 3.3.2 for details).

3.2.2 Orientation Term

For each IMU, i, an orientation constraint is added which seeks to minimize the relative orientation between the measured and solved global bone orientation (Fig. 2).

The measured global bone orientation, $\mathbf{R}_{b_i}^g$ is obtained from the IMU measurement $\mathbf{R}_i$ using the IMU-bone offset $\mathbf{R}_{ib}$ and IMU reference frame-global offset as follows:

$$\mathbf{R}_{b_i}^g = \mathbf{R}_{ig} \cdot \mathbf{R}_i \cdot (\mathbf{R}_{ib})^{-1} . \tag{8}$$

The solved global bone orientation, $\hat{\mathbf{R}}_b^g$ is obtained using the kinematic chain, ignoring translations:

$$\hat{\mathbf{R}}_b^g = \prod_{b' \in \mathcal{P}(b_i)} \mathbf{R}_{b'} . \tag{9}$$

and the orientation cost is

$$E_R(\boldsymbol{\theta}) = \sum_{i \in [1, n_i]} \rho_R \left(\lambda_R \left\| \boldsymbol{\psi} \left((\hat{\mathbf{R}}_{b_i}^g)^{-1} \mathbf{R}_{b_i}^g \right) \right\|_2^2 \right) \tag{10}$$

where $\boldsymbol{\psi}(\cdot)$ extracts the vector part of the quaternion representation of the rotation matrix, λ_R is orientation constraint weighting factor, $\rho_R(\cdot)$ is a loss function. Discussion of the weightings and loss functions are deferred to Sect. 3.2.6.

3.2.3 Acceleration Term

In addition to orientation, the IMUs provide acceleration measurements (in the local IMU coordinates). In order to include an acceleration term, it is necessary to consider a window of three frames: current frame to be solved, t, and the previous two solved frames, $t-1$ and $t-2$. For each IMU, a constraint is added which seeks to minimize the difference between the measured and solved acceleration of the track target site (Fig. 2). The solved acceleration $\hat{\mathbf{a}}_i^g$ is computed by central finite differences using the solved pose from previous two frames along with the current frame:

$$\hat{\mathbf{a}}_i^g(t-1) = \left(\hat{\mathbf{t}}_i^g(t) - 2\hat{\mathbf{t}}_i^g(t-1) + \hat{\mathbf{t}}_i^g(t-2) \right) / (\Delta t)^2 . \tag{11}$$

where the solved IMU positions $\hat{\mathbf{t}}_i^g$ are computed analogously with Eq. 5 (replacing subscripts p with i) and Δt is the frame period.

The measured local accelerations from the previous frame of IMU data[1] are converted to global coordinates as follows:

$$\mathbf{a}_i^g(t-1) = \mathbf{R}_{ig} \cdot \mathbf{R}_i(t-1) \cdot \mathbf{a}_i(t-1) - \mathbf{a}_g \tag{12}$$

where $\mathbf{a}_g = [0, 9.8707, 0]^T$ is the acceleration of gravity, which needs to be subtracted. The acceleration cost is then simply defined as

$$E_A(\boldsymbol{\theta}) = \sum_{i \in [1, n_i]} \rho_A \left(\lambda_A \left\| \hat{\mathbf{a}}_i^g - \mathbf{a}_i^g \right\|_2^2 \right) \tag{13}$$

where once again λ_A is a constraint weighting factor, $\rho_A(\cdot)$ is a loss function (see Sect. 3.2.6).

Note that the orientation constraints only require the orientation offset of the IMU w.r.t. the bone to be known, whereas

[1] The previous frame is used because central differences are employed to estimate the solved acceleration.

the acceleration constraints require the translational offset to be known as well.

It is well known that double integrating acceleration to obtain position is prone to drift, thus these acceleration terms alone are not sufficient to locate the body in global coordinates over any length of time. The acceleration term is included in the method for completeness, however in the experiments, it was found not to improve performance.

3.2.4 PCA Prior Terms

In practice, not all the body segments are observed in the input - the kinematic skeleton has more degrees of freedom than are constrained by the IMUs and positional measurements. For instance, the spine has several segments, but has only one or two IMUs attached to it. A pose prior is therefore required to constrain all degrees of freedom and produce plausible poses in spite of sparse or noisy sensor input.

In these experiments, two prior terms are incorporated based on a principal component analysis (PCA) of a corpus of motion capture data. The pose prior should be invariant to the global position and heading of the subject. We therefore use $\bar{\boldsymbol{\theta}}$, denoting the $d_p = d - 6$ pose vector excluding the first six elements, in the pose prior formulation.

A subset of ground-truth motion sequences from the *Total Capture* dataset (Trumble et al. 2017), covering a wide variety of poses were used as training of the PCA pose model. In order to obtain a representative sample of poses without over-emphasis on commonly recurring poses for standing and walking, for instance, we perform k-means clustering on the full set of $n_f = 126{,}000$ training frames, with $k = n_f/100 = 1260$. The cluster centres are concatenated to form a $k \times d_p$ data matrix $\mathbf{D}$ and PCA is performed on the mean-centered data. The dimensionality is reduced to $d_r = 23$ (chosen so as to keep 95% of the variance in the data) and the resulting PCA model is a $d_p \times d_r$ coefficient matrix, $\mathbf{M}$, a d_p-dimensional mean vector, $\boldsymbol{\mu}$ and a d_r-dimensional vector of standard deviations, $\boldsymbol{\sigma}$ (the square-roots of the principal component eigenvalues).

Similar to Ichim and Tombari (2016), we use two priors based on the PCA of the pose: PCA projection and PCA deviation. The projection prior encourages the solved body pose to lie close to the reduced dimensionality subspace of prior poses (soft reduction in the degrees of freedom of the joints), while the deviation prior discourages deviation from the prior observed pose variation (soft joint rotation limits). The pose projection cost is

$$E_{PP}(\boldsymbol{\theta}) = \rho_{PP}\left(\lambda_{PP}\left\|(\bar{\boldsymbol{\theta}} - \boldsymbol{\mu}) - \mathbf{M}\mathbf{M}^T(\bar{\boldsymbol{\theta}} - \boldsymbol{\mu})\right\|_2^2\right) \quad (14)$$

and the pose deviation cost is

$$E_{PD}(\boldsymbol{\theta}) = \rho_{PD}\left(\lambda_{PD}\left\|\mathrm{diag}(\boldsymbol{\sigma})^{-1}\mathbf{M}^T(\bar{\boldsymbol{\theta}} - \boldsymbol{\mu})\right\|_2^2\right) \quad (15)$$

where as with the data terms, weighting factors λ and loss functions ρ are used (see Sect. 3.2.6). A geometric interpretation of these constraints is shown in Fig. 2. Together these terms produce soft constraints that yield plausible motion while not strictly enforcing a reduced dimensionality on the solved pose, thus allowing novel motion to be more faithfully reproduced at run time.

3.2.5 Floor Penetration Term

For indoor operation on a level surface, we are able to include a simple constraint which disallows solved positions of joints being below the floor level:

$$E_{FP}(\boldsymbol{\theta}) = \sum_{p \in p_e} \lambda_{FP}\left\|H(-[\hat{\mathbf{t}}_p^g]_y + r_p)\right\|_2^2 \quad (16)$$

where r_p is a collision radius used to approximate the distance between the joint and the surface (set to 5 cm in our experiments), the operator $[]_y$ extracts the y component of a vector, H is the Heaviside step function and p_i are the set of targets (joint centres) on for which the constraint is enabled (see Fig. 2). This constraint is used for indoor capture scenarios in which the floor level is known from the camera calibration, during which the world coordinates are centred at floor level with the y-axis up. Note that this term is not applicable to outdoor scenes which feature uneven terrain, and is thus disabled for all our outdoor experiments. In our experiments the constraint is added for the feet and toe joints only in order to keep computation time low.

3.2.6 Energy Minimization

As described in the previous subsections, weightings λ are used to control the contributions of each term to the overall cost in Eq. 2. These are required because the different terms compare different physical quantities, and because some sources of data may be more reliable than others - for instance IMU orientations may be more stable than noisy position triangulations from images. Throughout the experiments, the same weightings were used for the cost function terms, namely $\lambda_P = 1 \times 10^{-3}$, $\lambda_R = 1$, $\lambda_A = 1 \times 10^{-3}$, $\lambda_{PP} = 0.7$, $\lambda_{PD} = 0.06$ and $\lambda_{FP} = 10$. These values were arrived at by a gradient-based parameter optimization over 200 frames of one motion sequence (S1, FS1) from the *Total Capture* dataset.

Furthermore, each term has a loss function, $\rho(\cdot)$ for each residual. The purpose of the loss function is to make the cost robust against outlier data (as well as to allow deviation from the prior, when the measurements support it). For

the orientation constraints, a null loss is used (standard L2 distance), since the IMUs tend not to produce outlier measurements. For the position, acceleration, PCA projection prior and PCA deviation prior a robust Cauchy loss function is used, $\rho(x) = \log(1 + x)$. The Cauchy loss function limits the effect of gross outliers by penalizing large residual values proportionally less than small values. Using the robust loss functions was found to be necessary to get good pose estimations in the presence of outlier measurements as well as novel, unseen poses.

The pose cost function $E(\boldsymbol{\theta})$ is optimized using the Ceres non-linear least-squares solver (Agarwal et al. 2017). The position, orientation and acceleration constraints are only affected by parameters associated with the bone to which they are attached and its parent bones in the kinematic chain. Therefore, the Jacobian is sparse and its computation can be sped up by using parameter blocks. The computation is further sped up using multi-threaded Jacobian computation. The solving is performed using Levenberg–Marquardt with a sparse normal Cholesky linear solver. For each frame, the pose vector is initialized with the solved value from the previous frame, yielding full-body 6-DoF pose estimation at real-time video rates.

3.3 2D Keypoint Detections from Multi-view Video

The convolutional pose machines (CPMs) detector of Cao et al. (2017) as implemented in OpenPose[2] is used to obtain 2D keypoint detections $\mathbf{t}_p^c$. The detector also outputs confidences, w_p^c. The detections are used in the positional constraints for each view in the cost function (Sect. 3.2.1) for each subject. Section 3.3.1 describes our approach to resolving and assigning unordered input 2D detections to each tracked subject.

Whereas Malleson et al. (2017) use 12 joint keypoints from the 15 keypoint MPI model in OpenPose (shoulders, elbows, wrists, upper legs, knees and ankles), we use the 'Body25' model (which includes the above as well as the mid hip, nose, eyes, ears, neck, big toes, small toes and heels). In our experiments we found the 'Body25' model to produce more accurate results (with reduced processing time) compared to the 'MPI15' and 'COCO18' models also available in OpenPose. In Sect. 3.3.2, we describe a procedure for calibrating 3D offsets between the CPM detector keypoints and the corresponding bones in the kinematic skeleton. Note that our approach is agnostic as to the source of the 2D keypoint detections and any suitable 2D pose keypoint detector could be used in place of the CPM detector.

In order to increase detection throughput and maintain real-time frame-rates with multiple cameras, we pack multiple camera views into a single frame for detection (refer to Sect. 3.4 for details of the live implementation).

3.3.1 Detection Sorting

In order to track multiple subjects simultaneously and to be robust against any incidental 2D detections in the images (caused by bystanders or spurious detections), we sort the 2D detections as described below. In our approach, only a designated set of subjects are tracked and their initial locations are set manually (e.g. through having the subjects begin at designated locations in the capture space). Any other subjects in the scene should be designated as bystanders and rejected by the detection sorter.

The CPM-based 2D keypoint detector of Wei et al. (2016) produces a set of 2D keypoints with confidences for each detected person in each camera viewpoint C. For brevity, we refer to each of these as a '2D person', P. In order to use these as input in our multi-person 3D tracking approach, the correct 2D person must be assigned to each tracked subject S. The order of the raw input 2D person detections is arbitrary and generally not consistent between cameras or over time. In general wide-baseline camera setups, assumptions such as left/right consistency cannot be used. Moreover, a given subject may be absent in a view due to occlusion, and spurious detections may occur due to mis-detections or bystanders being visible in the scene. In order to use the 2D person detections in our framework, they first need to be sorted and assigned to the corresponding tracked subject or discarded if they do not belong to any of the tracked subjects.

The procedure for sorting the 2D person detections over all subjects and camera views is presented in Algorithm 1. In summary, all pairings of 2D person detections over each pair of camera views are putatively triangulated and checked for consistency in re-projection. Valid candidates are then matched to the expected 3D locations of each tracked subject or discarded if no match is found.

First, a set of 3D person candidates $CandsAll$ is initialized (line 1). This is used to store source camera and 2D person indices as well as triangulated 3D positions for subject candidates. (Note that these putative 3D triangulations are only used in the detection sorting stage and not in the kinematic pose optimization, Eq. 2, which uses the sorted 2D keypoints directly and does not employ explicit triangulation.)

Triangulation is attempted across all pairs of 2D people across all pairs of views to obtain a candidate 3D person P_{3D} (line 2–6). Note that only keypoints with non-zero confidences in both 2D person detections are considered. Candidates which have an average re-projection error above a threshold $ThreshTriError$ are rejected as these are likely to come from different people (line 7). Note that this threshold needs to be set relatively high, due to levels of noise in the 2D detections. However, if set too high, a high level

[2] The implementation used is available from: https://github.com/CMU-Perceptual-Computing-Lab/openpose.

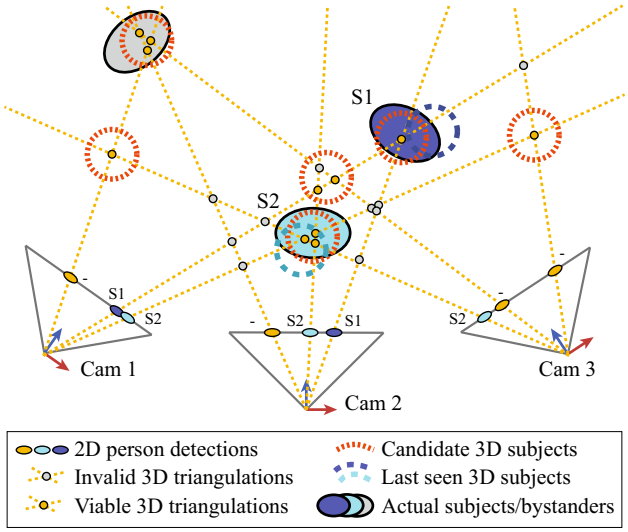


Fig. 3 Visualization of the proposed approach to sorting and assigning 2D detections to tracked subjects. All possible pairings of detected person instances across views are tested for image re-projection error. Triangulations with high re-projection error are regarded as invalid, while viable triangulations are clustered into candidate 3D subjects. Finally, each tracked subject is assigned to the candidate 3D subject closest to its last seen location

of invalid matches occurs. In the experiments, an empirical value of 1.5% of the image width was used.

Next, P_{3D} is compared to all candidates already in $CandsAll$. If on average distance between the keypoints in P_{3D} and those in any existing candidate $CandEx$ is less than a threshold $ThreshDistSamePerson$ (empirically set to 30 cm in the experiments), P_{3D} is assumed to correspond to the same person and the current source indices and 2D person are appended to $CandEx$ (line 8–9), otherwise they form a new candidate, $CandNew$, which in turn is appended to $CandsAll$ (line 10–12).

The candidate 3D detections $CandsAll$ are assigned to the tracked subjects S. In order to prevent assignment of the same detections to more than one subject, a condition variable $CandAssigned$ is maintained to keep track of which candidates have already been assigned to a subject (line 19–21). The sorted detections are initialized empty (line 23). Finally, the closest un-used candidate in $CandsAll$ to the last viewed location of the subject S, $Subjects3DLast$ is determined and $SortedDetections(S)$ set if it is within a threshold $ThreshInterframeMovement$ (empirically set to 1 m in the experiments), the assumed maximum distance between the current and last viewed position of the subject (line 24–30). A conceptual visualization of the sorting approach is presented in Fig. 3. Note that the detection sorter requires at least two input cameras due to the requirement for 3D triangulation, thus our approach cannot currently handle multiple subjects in the case of monocular input.

Algorithm 1 Sorting of 2D detections into subjects for a single frame. Takes as input all 2D person detections P from all camera views C and attempts to assign each 2D detection to its corresponding 3D tracked subject S while disregarding any non-matching detections (e.g. due to bystanders in the scene).

1: $CandsAll \leftarrow []$
2: **for** all cams $C_a : a \in [1, n_c]$ **do**
3: **for** all 2D people P_a in C_a **do**
4: **for** all cams $C_b : b \in [a+1, n_c]$ **do**
5: **for** all 2D people P_b in C_b **do**
6: $P_{3D} \leftarrow$ Triagulate(P_a, P_b)
7: **if** TriError(P_{3D}) $< ThreshTriError$ **then**
8: **if** $\exists$ $CandEx \in CandsAll$ such that AveDist($CandEx$, P_{3D}) $< ThreshDistSamePerson$ **then**
9: $CandEx \xleftarrow{+} [(C_a, P_a), (C_b, P_b), P_{3D}]$
10: **else**
11: $CandNew \leftarrow [(C_a, P_a), (C_b, P_b), P_{3D}]$
12: $CandsAll \xleftarrow{+} CandNew$
13: **end if**
14: **end if**
15: **end for**
16: **end for**
17: **end for**
18: **end for**
19: **for** all candidates i in $CandsAll$ **do**
20: $CandAssigned(i) \leftarrow false$
21: **end for**
22: **for** $S \in [1, n_s]$ **do**
23: $SortedDetectons(S) \leftarrow []$
24: $[i_{best}, d_{best}] \leftarrow$ ClosestPersonInSet($CandsAll$, $Subjects3DLast(S)$)
25: **if** $d_{best} < ThreshInterFrameMovement$ and $CandAssigned(i_{best}) = false$ **then**
26: $SortedDetectons(S) \leftarrow CandsAll(i_{best})$
27: $CandAssigned(i_{best}) = true$
28: $Subjects3DLast(S) \leftarrow CandsAll(i_{best})$
29: **end if**
30: **end for**

3.3.2 Positional Track Target Offset Calibration

As described in Sect. 3.1, the positional track targets, p are defined by a translational offset $\mathbf{t}_{pb}$ w.r.t. their corresponding bone b. Whereas *surface* keypoints (e.g. ears, eyes) should clearly exhibit offsets from the joints of their corresponding bones in the skeleton, *joint* keypoints (e.g. knees, elbows) should theoretically not be offset w.r.t. the bone joint (i.e. $\mathbf{t}_{pb} = \mathbf{0}$). In practice, however, systematic differences in reference and estimated joint position can and do arise, for instance due to inaccurate calibration of subject-specific optical mo-cap skeletons, or the characteristics of the data used to train the detector. In Malleson et al. (2017), only joint keypoints were used and the assumption of zero offset w.r.t. the skeleton bones was made. In this work, we attempt to mitigate systematic bias caused by this by estimating calibrated offsets between the CPM detected keypoints and our 'ground

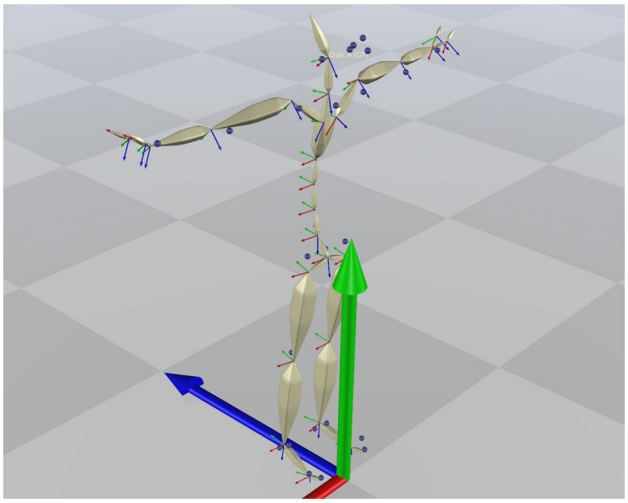


Fig. 4 Visualization calibrated positional track targets (blue) offsets for the CPM Body25 keypoint set w.r.t. the optical reference skeleton (yellow). Note the offset for both the surface and the joint keypoints (Color figure online)

truth' optical reference skeleton. As is shown in Sect. 4.1.2, including these calibrated offsets in the solve reduces the output error substantially.

The following procedure is used to estimate constant bone-space offsets $\mathbf{t}_{pb}$ for all keypoints p. For a given input frame, for each keypoint, p, a confidence-weighted multi-view triangulation across all cameras is performed to obtain a global 3D position estimate, $\hat{\mathbf{t}}_{pg}$:

$$\hat{\mathbf{t}}_{pg} = \underset{\mathbf{t}_{pg}}{\operatorname{argmin}} \sum_{c\in[1,n_c]} \rho_P\left(w_p^c \left\| \mathrm{dh}(\mathbf{P}_c \mathbf{t}_{pg}) - \mathbf{t}_p^c \right\|_2^2\right). \tag{17}$$

This is converted to bone-local coordinates using the ground truth global pose of the bone, $\mathbf{T}_b^g$:

$$\hat{\mathbf{t}}_{pb} = (\mathbf{T}_b^g)^{-1} \hat{\mathbf{t}}_{pg} \tag{18}$$

Finally, $\mathbf{t}_{pb}$ is taken as the per-axis median of $\hat{\mathbf{t}}_{pb}$ over all input frames, ensuring robustness against outlier detections. Figure 4 illustrates the calibrated offets on our solving skeleton, for both surface and joint keypoints, with the calibrated offsets up to 6.5 cm from the joint centre.

3.4 Implementation Details

3.4.1 Input Data

Xsens MTw wireless IMUs (Roetenberg et al. 2013) were used for all our datasets as well as the live demonstrator system. Each MTw IMU contains gyroscopes, accelerometers and magnetometers and through internal onboard sensor fusion outputs an orientation and acceleration at 60 Hz. The inertial reference frame of each IMU, $\mathbf{R}_{ig}$ is assumed to be consistent between IMUs and in alignment with the world coordinates through the global up direction and magnetic north. The IMU-bone positions $\mathbf{t}_{ib}$ are specified by manual visual alignment and the IMU-bone orientations $\mathbf{R}_{ib}$ are calibrated using the measured orientations with the subject in a known pose (the T-pose, facing the direction of a given axis). To temporally align the IMU and video data an initial footstamp was performed by the actor, which is visible in the video and produces a strong peak in acceleration in the IMU data.

The camera intrinsics and extrinsic calibration is determined using a checker-board chart (after Zhang 1999) and for simplicity of integration with the inertial measurements, the global reference frame of the camera system is chosen to lie on the floor and to align with the up direction and magnetic north. The number of input cameras, their resolution and their frame-rate vary between datasets (between 0.5 and 4 k pixels wide and from 30 to 60 fps). The CPM detector, however, operates on relatively low-resolution images (656 × 368), and the input is downscaled accordingly.

3.4.2 Live Implementation

Here we briefly describe a portable live implementation of the proposed real-time motion capture approach. Note that, unless otherwise specified, the same core implementation and processing parameters are used in all experiments (live or offline). The live system differs from the offline results only that the input video frames are temporally sub-sampled and the detections interpolated online so as to maintain live operation, whereas for the (recorded) datasets, detection is performed on all input frames.

The capture and pose solving of one or more subjects is performed in real time on a single commodity laptop PC (Intel i7 3.6 GHz with 32 GB RAM, NVIDIA GTX 1080 mobile GPU with 8GB RAM) using multi-threaded C++ code. The laptop is connected to 4 PointGrey *Grasshopper 3* machine vision cameras (each capturing at 472 × 512, 60 fps), and optionally to 1 or 2 Xsens receivers (with up to 17 Xsens MTw wireless IMUs for per subject). The solved motion of each character may be recorded or streamed live to a game engine environment (e.g. Unity or Unreal Engine) on a second laptop, which displays the game environment and character in VR (see Fig. 15 and the supplementary video).

Because hardware synchronization is not available between camera and IMU inputs and because the GPU-accelerated CPM-based detector runs at a lower rate than our kinematic solver, it is not possible to process with fixed frame intervals while maintaining live operation. The system is therefore run asynchronously, with a fixed delay between output solver time and 'wall' time. Buffers of the 2D keypoint detections and IMU input are sampled and interpolated according to

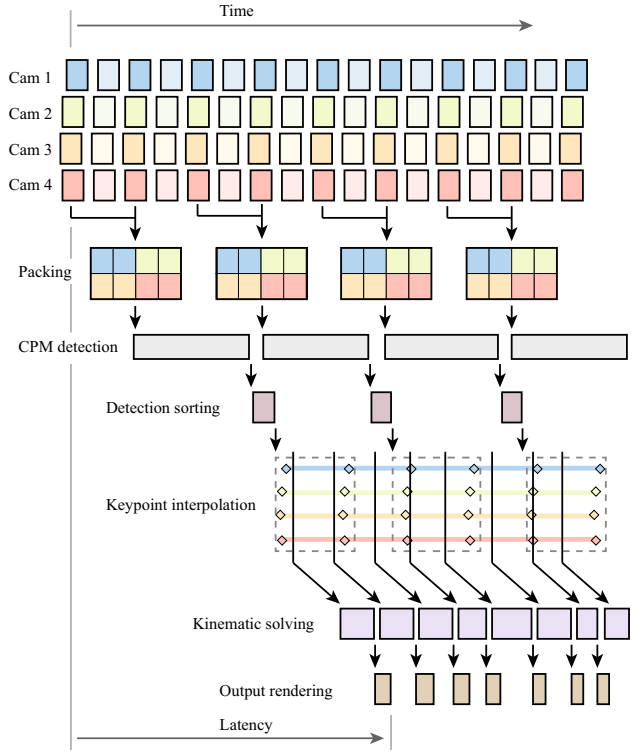


Fig. 5 Processing flow for live operation showing 4 camera capture, packing multiple frames into single images, CPM detection, detection sorting and interpolation for input into the kinematic solve (timing not shown to scale). Note that the cameras are not assumed to be synchronised, and that the cameras may operate at a higher frame-rate than the CPM processing achieves, as illustrated here by the alternate greyed-out frames in the capture sequence

the current solve time (linear interpolation is used for the 2D keypoint detections and quaternion interpolation for the IMU orientations).

To increase the through-put of the CPM-based 2D keypoint detection, 2 frames of each of the 4 cameras are packed into a single image for detection (the CPM detection time is independent of the number of subjects in the image, see Cao et al. 2017). The live processing pipeline is illustrated in Fig. 5, showing the main tasks of multi-camera capture, keypoint detection and kinematic solving being run in parallel, with the current solver time delayed by a fixed latency.

3.4.3 Timing

On our system, the typical CPM detection time is approx. 110 ms per image, giving an effective 2D detection rate of approx. 18 fps (compared to approx. 2.3 fps with sequential detection of frames). Since the GPU is only approx. 60% utilized, further increases in detection frame-rate might be achieved in future work by using two CPM detectors in parallel.

The solved kinematic pose is output at approx. 40 fps with a single subject and approx. 25 fps with two subjects, making it well suited to real-time media production and interactive applications. The end-to-end latency of the system is approx. 230 ms.

4 Results and Evaluation

The proposed approach is designed to take multi-view video and IMUs as input. It is flexible in terms of the number of input cameras and IMUs, degrading gracefully as the number of cameras and IMUs is reduced and still functioning in extreme cases such as using a single (monocular) camera with no IMUs or one modality without the other. Note that in this work, all positional constraint information is obtained from the multiple-view video based on per-view CPM as discussed in Sect. 3.3 and no optical markers or visible targets are used.

We quantitatively evaluate our approach using the *Total Capture* dataset (Trumble et al. 2017), which features a single subject captured with multi-view video, IMU and a commercial optical mo-cap system for ground truth reference motion. To the best of our knowledge, *Total Capture* is the only dataset which contains multi-view video, IMU and optical reference data. For completeness, we also evaluate our approach on the widely used *Human 3.6M* dataset of Ionescu et al. (2014), which does not contain IMUs input.

In addition to the quantitative evaluation, we perform extensive qualitative evaluation on various multi-view video plus IMU datasets including the single-person *Total Capture Outdoor* dataset (Malleson et al. 2017) and three new multi-person datasets, *Ping Pong*, *Karate* and *Outdoor Duo*. These were captured with natural, loose clothing, and, in the case of *Outdoor Duo*, in unconstrained outdoor environments. The *Ping Pong* and *Outdoor Duo* datasets will be made available for research use upon publication.

Finally, we show results from our real-time, live implementation, which is portable and runs on a single laptop PC with 4 machine vision cameras and up to 13 IMUs per subject. The live tracked motion can be streamed to a game engine environment and displayed in a virtual reality headset.

4.1 Quantitative Evaluation

The *Total Capture* dataset includes five subjects (*S*) performing various motions including range of motion (*ROM*), walking (*W*), acting (*A*), and 'freestyle' (*FS*). These sequences vary in complexity and speed from slow ROM sequences to challenging sequences including fast motion and unusual poses such as crouching on the floor (see Fig. 6 and refer to the supplementary video). The subjects were recorded simultaneously using 13 Xsens MTw IMUs, 8 HD video cameras and a commercial infra-red motion capture system consisting of 16 cameras and a dense set of retro-reflecting markers

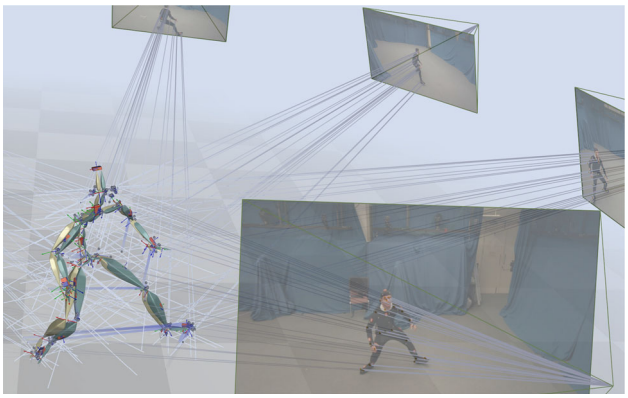


Fig. 6 Ground truth optical skeleton (yellow) and the solved skeleton (blue) overlaid along with visualizations of the CPM detections from each view back-projected with a line showing where each detection is pulling the corresponding track target. The proposed robust cost function yields a high-quality pose estimate despite several views containing mis-detections (e.g. swapped feet). Sequence: *Total Capture*, S5, FS1

worn by the subject. The marker-based input is not used in the runtime solver and is only used in this work as a 'ground truth' reference for evaluation.

Detailed results with the full input (8 cameras, 13 IMUs) are presented and compared to prior methods in Table 1, in which the proposed approach performs best on average with an average global joint position error of 26.1 mm, and average global joint orientation error of 7.5°.[3] Note that for this dataset, the proposed approach as well as the prior works compared against, are evaluated across all 21 solved joints in the kinematic skeleton (i.e. hip, four spine bones, neck, head, shoulders, elbows, wrists, uppers legs, knees, ankles and toe bases). In addition to mean global joint position and orientation errors, we report standard deviations as well as robust statistics, i.e. median and median absolute deviation (MAD). Furthermore, we include errors w.r.t. the position root joint (hips) of the subject, as is done by Trumble et al. (2017) and with procrustes alignment, giving an indication of the pose accuracy independent of errors in root position and orientation. Figure 6 shows the robustness of our approach to typical misdetections from the CPM joint detector.

4.1.1 Number of Cameras and IMUs

It is desirable to have a minimal capture hardware setup in order to reduce cost as well as actor setup time. We simulate the effect of reduced capture hardware (sparse input) by excluding selected cameras and IMUs from the input. For completeness, the special cases with no IMUs, no cameras, or a single camera, are included, effectively evaluating on IMU-only and monocular camera cases. The 13 IMUs in the full set are placed on the pelvis, sternum, upper and lower limbs, head and feet. The 6 IMUs in the reduced set are positioned on the pelvis, lower limbs and head. The full set of cameras form a ring around the subject and a the subsets used in the tests are the adjacent cameras starting from the first camera.

Figure 7 shows the error with 0–8 cameras along with 0, 6 or 13 IMUs. The results show show reasonable results can be obtained even with the minimal input configurations, including IMU only and monocular cases. The best results are obtained by combining IMU and multiple camera input. Using only 3 cameras is enough to provide accurate global position, with a more gradual improvement as further cameras are added. The sparse set of 6 IMUs provides substantial improvement in orientation error, while including all 13 IMUs improves the orientation error further. This demonstrates the flexibility of our approach in trading hardware requirements and set-up time for accuracy. Table 2 shows a comparison of our approach under the 0 camera, 6 IMU and 1 camera, 13 IMU cases to DIP (Huang et al. 2018) and VIP (von Marcard et al. 2018), respectively. Our approach yields comparable results on DIP while outperforming VIP, on the respective sensor configurations. A grid of visual results for two sequences across the camera/IMU sweep is shown in Fig. 8.

4.1.2 Cost Function Ablation Study

In order to assess the effectiveness of each term in the cost function, Eq. 2, we evaluate the solver with selected terms switched off. These results are summarized in Table 3, which shows the mean and standard deviation in solved joint position and orientation error with specific terms omitted from Eq. 2, relative to the error using the full cost function. This shows the effectiveness of fusing input from multiple modalities to obtain high quality pose estimates.

From the IMU input, the orientation term is effective in resolving the bone orientations, which are not fully defined by the joint locations. In addition the orientation term reduces the error in position, helping to mitigate jitter and misdetections present in the CPM keypoints. On the other hand, the acceleration term does not improve performance in the proposed formulation. With the optimized parameter weightings used, acceleration is weighted relatively low. As shown in Fig. 9, increasing the acceleration weighting by an order of magnitude, reduces the error in acceleration slightly, but ultimately increases the error in position. While the fusion of acceleration with positional information prevents long-term runaway drift, local overshooting occurs resulting in increased positional error and 'swimming' artefacts in the solved motion.

[3] In line with standard practice, the global joint orientation error is computed as the magnitude of the 3D rotation required to bring the solved and ground truth joint orientations into alignment (using the angle-axis norm).

Table 1 Results for 8 sequences from the *TotalCapture* dataset (Trumble et al. 2017), using input from all 8 cameras and 13 IMUs

Method seq.	S1, FS3	S2, FS1	S2, R3	S3, FS1	S3, FS3	S4, FS3	S5, A3	S5, FS1	Mean
Mean (std) pos. error (mm), global									
Malleson'17	74.0	53.0	39.0	67.0	64.0	67.0	74.0	70.0	63.5
Proposed	**36.1 (21.6)**	**16.1 (10.7)**	**12.8 (10.8)**	**27.1 (21.6)**	**26.2 (20.8)**	**40.0 (22.8)**	**20.9 (13.1)**	**29.2 (17.5)**	**26.1 (9.3)**
Mean (std) pos. error (mm), w.r.t. root									
Trumble'17	94.0	167.0	93.0	136.0	86.0	116.0	140.0	105.0	117.1
Trumble'18	39.5	**10.4**	21.3	**10.6**	63.7	87.4	33.6	17.3	35.5
Proposed	**31.7 (25.3)**	15.7 (11.9)	**13.7 (13.2)**	25.5 (25.5)	**23.4 (23.9)**	**26.5 (21.4)**	**19.3 (15.5)**	**24.4 (20.8)**	**22.5 (5.9)**
Mean (std) pos. error (mm), with procrustes									
Proposed	23.1 (16.5)	9.0 (6.4)	10.2 (8.7)	20.8 (18.8)	18.2 (19.6)	18.8 (13.7)	14.3 (8.6)	18.6 (13.2)	**16.6 (5.0)**
Mean (std) ori. error (deg), global									
Malleson'17	11.2	5.1	5.0	8.3	9.3	**8.0**	7.6	8.2	7.8
Proposed	**11.0 (8.3)**	**4.9 (3.7)**	**4.5 (4.3)**	**7.8 (7.3)**	**8.8 (8.2)**	8.1 (5.8)	**7.1 (5.8)**	**7.8 (5.8)**	**7.5 (2.1)**
Mean (std) ori. error (deg), with procrustes									
Proposed	10.9 (8.1)	4.6 (3.5)	4.6 (4.2)	7.8 (7.2)	8.8 (8.2)	8.0 (5.7)	7.1 (5.8)	7.7 (5.7)	**7.4 (2.1)**

Lowest errors are highlighted in bold

Mean error in absolute position (mm) and orientation (deg) over all joints and all frames is reported, as per the headings. Where available, standard deviations are shown in brackets (for the individual sequences, the standard deviation is computed across bones and frames, for the mean over all sequences (right most column), the standard deviation is calculated across sequences). Results are shown w.r.t. global coordinates, w.r.t. the root position, and with per-frame procrustes alignment. The results with procrustes alignment express the error in pose with the global rigid body pose component factored out. On average, our approach yields substantially lower positional error than previous approaches, while orientation is improved slightly

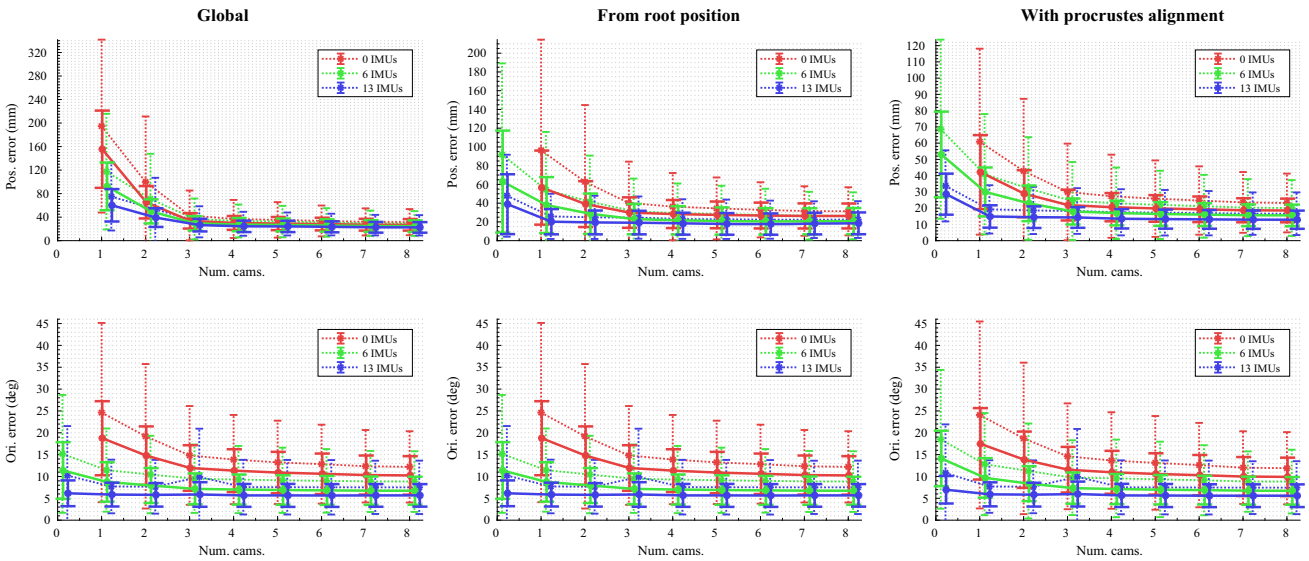


Fig. 7 Position and orientation error over 8 *Total Capture* sequences with different sensor configurations, with 0–8 cameras and 0, 6 and 13 IMUs. The dashed lines show mean and standard deviation in absolute position and orientation errors over all frames and bones, while the solid lines show the corresponding median and median absolute deviation (MAD). Results are measured w.r.t. global coordinates as well as w.r.t. the root position and with procrustes alignment. Reasonable results are achieved even with minimal input and high quality results are obtained with as few as three cameras

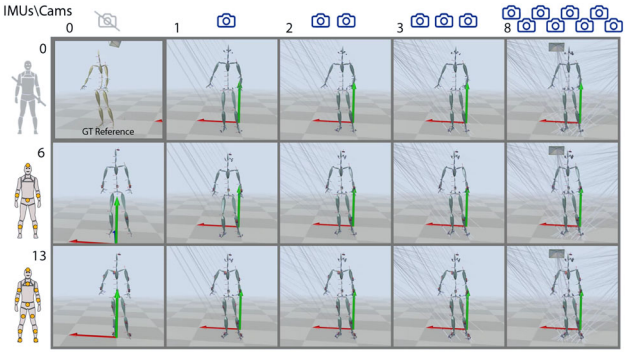
The position term is required in order to lock down the global position of the subject in 3D, thus an arbitrarily high global position error results without it. The local positional error as well as the orientation error are reduced by the position term.

The prior terms are effective in reducing the output error by encouraging plausible poses in the presence of noise in the input data and otherwise unconstrained degrees of freedom in the skeletal pose. Inclusion of the prior projection more

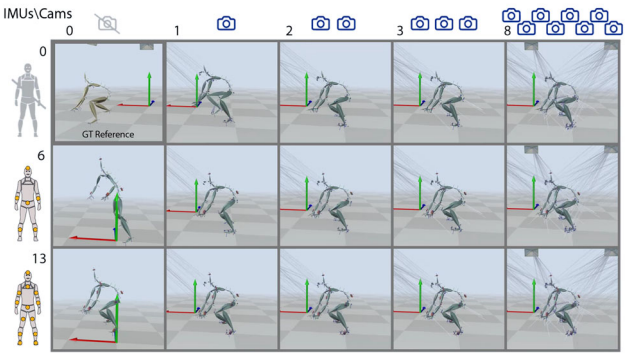
Table 2 Mean joint position and orientation error (with procrustes alignment) for the *Total Capture* dataset using sparse input data compared to DIP-online (Huang et al. 2018) and VIP (von Marcard et al. 2018)

Input config.	Approach	Mean (std) pos. error (mm)	Mean (std) ori error (deg.)
0 cam., 6 IMU	DIP-online	**59.6 (61.3)**	**15.8 (13.4)**
	Proposed	68.6 (55.1)	18.5 (15.9)
1 cam., 13 IMU	VIP	26.0	12.1
	Proposed	**19.2 (14.9)**	**7.7 (6.0)**

Lowest errors are highlighted in bold



(a) S1, FS1



(b) S5, FS1

Fig. 8 Visualization of results of the proposed approach for *Total Capture* S1, FS1 (**a**) and the more challenging S5, FS1 (**b**) over a range of input configurations. Note how the quality of the solved pose degrades gracefully as the input is made more sparse

than halves the error in position and orientation, while the prior deviation term results in a small improvement.

While the floor penetration term can improve the qualitative appearance of the solved motion (see Fig. 10) it does not significantly improve the numerical performance. Although it prevents the feet being reconstructed below the floor, the term does not necessarily cause the limbs to be reconstructed closer to their correct positions.

Finally, the last five rows of table show the effects of using improved input keypoints. Whereas Malleson et al. (2017) use a subset of 12 keypoints from the 'MPI15' detection model of OpenPose, the proposed approach uses all 25 keypoints available with the 'Body25' detection model of OpenPose (see Sect. 3.3) and furthermore applies calibrated offsets to the targets (Sect. 3.3.2). In all cases, the mean error in solved joint position is improved by using the calibrated offsets. Note that the same calibrated offsets are used for all subjects. (While the subjects in the *Total Capture* dataset range in height from 152 and 180 cm, scaling these offsets according to subject height was found not to significantly improve the results.) On the 12 keypoint subset, the body 'Body25' model provides a marginal improvement in accuracy, however a significant improvement is obtained by using all 25 keypoints with the 'Body25' detection model.

4.1.3 Human 3.6M Dataset

For completeness, we further test the proposed approach on the *Human 3.6M* dataset, which contains only four cameras and no IMU input. We follow the evaluation procedure used in prior works (e.g. Tome et al. 2017) reporting the mean joint position error across 17 joints (which correspond to those used on the *Total Capture* dataset, but without the four intermediate spine joints) and show results for each activity averaged over all takes of subjects S9 and S11, using global position for the multi-camera case and per-frame procrustes aligned position for the monocular case. As shown in Table 4, the proposed approach yields state-of-the-art performance using 4 camera input, while in the case of monocular input, the proposed approach is outperformed by specialised monocular pose estimation approaches.

We note that a simplistic baseline approach 'Tri-CPM' reported by Trumble et al. (2017), which consists of explicit 3D triangulation of OpenPose 2D keypoint detections across all camera views, performs poorly compared to the proposed approach, which is able to make better use of such 2D detections through its use of a robust kinematic solve.

Note that we do not perform any re-training of our PCA pose prior or changing of cost function term weightings for these experiments, the pose prior created from the *Total Capture* dataset is used throughout all our experiments. The bone offsets, however, are re-calibrated for this dataset, as we found significant differences between our ground truth skeleton and that of *Human 3.6M* (refer to Sect. 3.3.2). Figure 10 shows our results on the 'Directions' and 'Sitting Down' sequences. The floor penetration term can improve the perceptual quality of the results since visually obvious pose errors in the form of the legs being reconstructed under the floor are eliminated, however, since error within the floor plane is still possible, the numerical error is not necessarily improved by the term (on this dataset, results are marginally worse by approx. 2% when the floor penetration term is disabled).

Table 3 Mean and standard deviations in solved joint position and orientation error with specific terms omitted, relative to the error using the full cost function, Eq. 2

Cost terms	Relative mean (std) pos. error		Relative mean (std) ori. error	
	Global	Local (procrustes)	Global	Local (procrustes)
Full (Eq. 2)	1.00 (1.00)	1.00 (1.00)	1.00 (1.00)	1.00 (1.00)
Without IMU	1.14 (1.20)	1.33 (1.35)	1.57 (1.34)	1.59 (1.39)
Without E_R	1.17 (1.25)	1.37 (1.40)	1.60 (1.36)	1.61 (1.41)
Without E_A	0.99 (0.98)	0.98 (0.99)	1.00 (1.00)	1.00 (1.00)
Without E_P	49.0 (36.5)	1.33 (1.44)	1.09 (1.08)	1.11 (1.17)
Without prior	1.86 (2.00)	2.55 (2.58)	3.14 (5.91)	3.32 (5.99)
Without E_{PP}	1.33 (1.37)	2.50 (2.77)	2.25 (2.61)	2.48 (3.15)
Without E_{PD}	1.02 (1.04)	1.06 (1.11)	1.16 (1.47)	1.18 (1.49)
Without E_{FP}	1.00 (1.00)	1.00 (1.00)	1.00 (1.00)	1.00 (0.99)
M15/K12 no calib.	1.29 (1.15)	1.23 (1.17)	1.09 (1.06)	1.07 (1.06)
M15/K12 calib.	1.18 (1.14)	1.10 (1.19)	1.04 (1.02)	1.03 (1.02)
B25/K25 no calib.	1.22 (1.14)	1.24 (1.11)	1.06 (1.02)	1.07 (1.02)
B25/K12 no calib.	1.23 (1.12)	1.09 (1.07)	1.05 (1.02)	1.03 (1.02)
B25/K12 calib.	1.17 (1.10)	1.04 (1.10)	1.02 (1.00)	1.01 (1.00)

The last five rows show the relative error when using different variations of input keypoints: the 12 keypoint subset (K12) from the 'MPI15' detection model (M15), the 'Body25' model without calibrated offsets (B25), and using the 12 keypoint subset of the 'Body25' detection model (B25/K12) (results averaged over S1 FS1 and S2 FS1 of *Total Capture* dataset with 8 cameras, 13 IMUs)

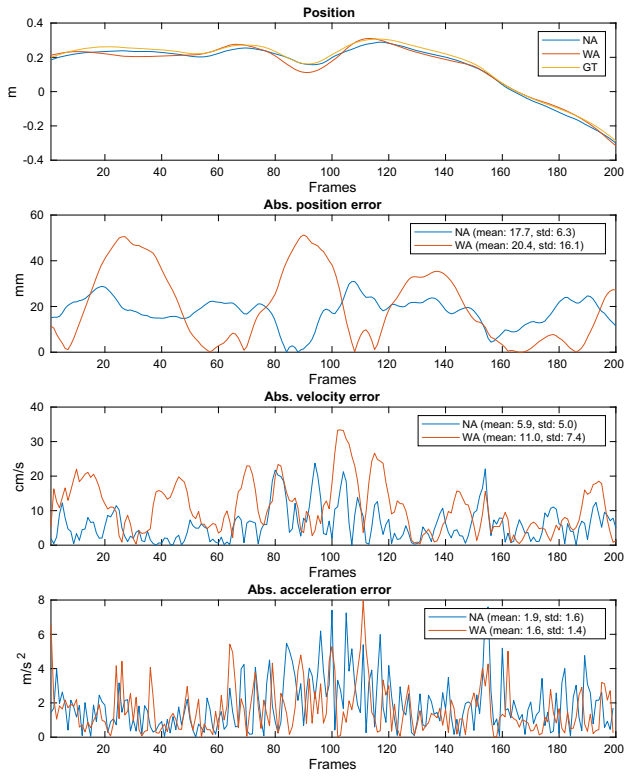
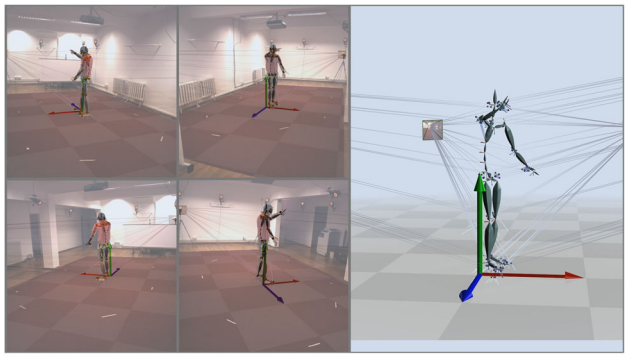
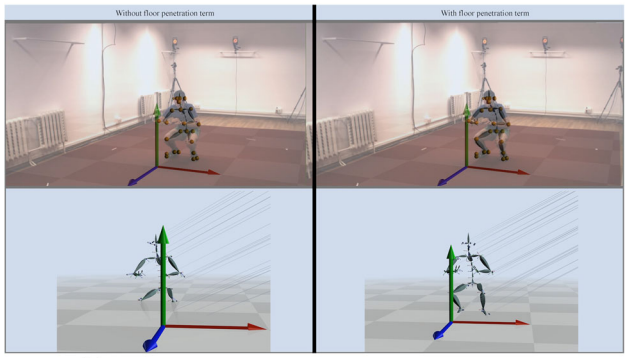


Fig. 9 Plots of the z-axis component of the root position, and corresponding absolute error in position, velocity and acceleration with the default low weighting of the acceleration term, $\lambda_A = 1 \times 10^{-3}$ (NA), and with an increased weighting, $\lambda_A = 1 \times 10^{-2}$ (WA). Note that while increasing the weight of the acceleration term results in a decrease in the mean error in acceleration, it causes increases in the mean errors in velocity and position, with the solved position exhibiting overshooting artefacts. Sequence: *Total Capture*, S1, FS1



(a) S9, Directions (4 cameras, no IMUs)



(b) S11, Sitting Down (monocular, no IMUs)

Fig. 10 Results on the *Human 3.6M dataset*. **a** High-quality results with 4 camera input. **b** Monocular solve without (left) and with (right) the floor penetration term enabled, the latter avoiding objectionable floor penetration artifacts

Table 4 Evaluation of mean joint position error (mm) for all sequences for test subjects S9 and S11 of the *Human 3.6M* dataset (Ionescu et al. 2014), left: 4 cameras, video input only, right: monocular video only

Approach	4 cameras (global)								Monocular (local - procrustes)			
Seq.	Ionescu	Tri-CPM	Tekin	Tome'17	Trum'17	Trum'18	Tome'18	Proposed	Lin	Martinez	Tome'18	Proposed
Directions	132.7	125.0	85.0	65.0	92.7	41.7	43.3	**38.5**	58.0	39.5	**38.2**	65.5
Discussion	183.6	111.4	108.8	73.5	85.9	43.2	49.6	**42.1**	68.3	**43.2**	40.2	69.9
Eating	132.4	101.9	84.4	76.8	72.3	52.9	**42.0**	43.5	63.3	46.4	**38.8**	94.8
Greeting	164.4	142.2	98.9	86.4	93.2	70.0	48.8	**42.1**	65.8	47.0	**41.7**	63.2
Phoning	162.1	125.4	119.4	86.3	86.2	64.9	**51.1**	56.2	75.3	51.0	**44.5**	109.8
Photo	205.9	147.6	95.7	110.7	101.2	83.0	64.3	**50.2**	93.1	56.0	**54.9**	92.5
Posing	150.6	109.1	98.5	68.9	75.1	57.3	**40.3**	41.5	61.2	41.4	**34.8**	72.3
Purchases	171.3	133.1	93.8	74.8	78.0	63.5	43.3	**42.5**	65.7	40.6	**35.0**	61.7
Sitting	151.6	135.7	73.8	110.2	83.5	61.0	66.0	**60.8**	98.7	56.5	**52.9**	105.6
Sit. Down	243.0	142.1	170.4	173.9	94.8	95.0	95.2	**77.7**	127.7	**69.4**	75.7	150.1
Smoking	162.1	116.8	85.1	85.0	85.8	70.0	**50.2**	54.8	70.4	49.2	**43.3**	94.7
Waiting	170.7	128.9	116.9	85.8	82.0	62.3	**52.2**	46.9	68.2	**45.0**	46.3	78.1
Walking	96.6	105.2	62.1	71.4	94.9	53.7	51.1	**50.0**	50.6	49.5	**44.7**	74.6
Walk. Dog	177.1	111.2	113.7	86.3	114.6	66.2	**43.9**	53.6	73.0	38.0	**35.7**	71.4
Walk. Together	127.9	124.2	94.8	73.1	79.7	**52**.4	45.3	46.9	57.7	43.1	**37.5**	71.5
Mean	162.1	124.0	100.1	88.5	88.0	62.5	52.8	**49.8**	73.1	47.7	**44.6**	85.0

Lowest errors are highlighted in bold
Note that IMU input is not available for this dataset. On average, The proposed approach out-performs previous approaches when using multi-camera input

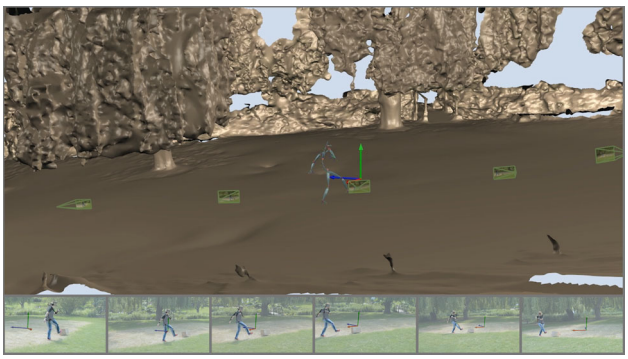


Fig. 11 Results on the single-person outdoor dataset *Total Capture Outdoor, Props*

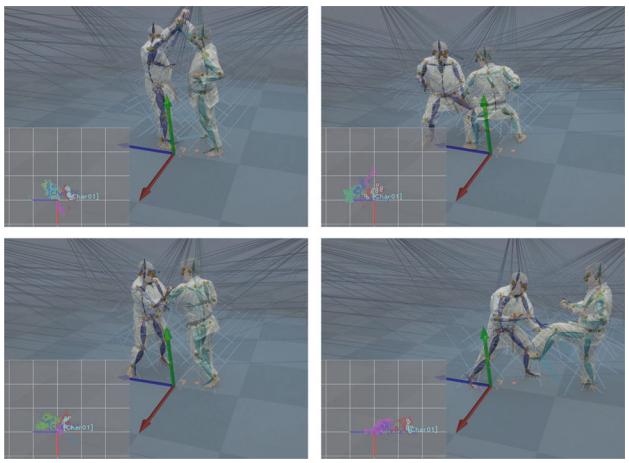


Fig. 12 Results for four frames of the two-person sequence *Karate, Action 13* (8 cameras, 2 × 13 IMUs per subject). Inset are top view visualizations of the subject sorter showing 3D candidates and labelled sorted subjects

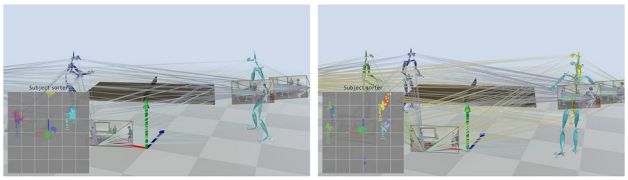
4.2 Qualitative Evaluation

4.2.1 Single Subject

The *Total Capture Outdoor* dataset (Malleson et al. 2017) was recorded outdoors in challenging uncontrolled conditions with a moving background and varying illumination. A set of 6 cameras were placed in a 120° arc around the subject and 13 Xsens IMUs. No ground truth data is available for this dataset. Figure 11 shows the 3D solved motion overlaid on each input view. The background models for this and the other outdoor sequences are generated from photogrammetry for visualization purposes. This background geometry is not used in the solver. Additional sequences for this dataset are presented in the supplementary video.

4.2.2 Multiple Subjects

The multi-person capability of the proposed approach was tested on two indoor datasets *Karate* (Fig. 12) and *Ping Pong* (Fig. 13) as well as on a new outdoor dataset *Outdoor Duo*,



(a) 2 subject (P5) **(b)** 4 subject (P4)

Fig. 13 Results on the *Ping Pong* dataset, sequences P4 and P5 (6 cameras, 0 IMUs)

which will be released for research use along with this paper. Details of the datasets are provided in Table 5.

These sequences are diverse in terms of environment and activities performed and include dynamic backgrounds, fast motion, close interaction, occlusion, and props. Results for four of the *Outdoor Duo* sequences are shown in Fig. 14. Finally, results from the live implementation with two subjects are shown in Fig. 15. Refer to the supplementary video for the full sequences and further visualizations.

Due to the ambiguity and imprecision of triangulation with approximate keypoints, the proposed detection sorting approach occasionally misassigns detections to subjects resulting in tracking failure. An example of such a failure case is shown in the supplementary video, in which the subjects are swapped following a close interaction. Suggestions for improving the robustness of the sorter are presented in the conclusion.

5 Conclusion

The proposed approach is able to obtain high-quality pose of one or more subjects in real-time. It is flexible in terms of camera and IMU hardware requirements, degrading gracefully as the number of cameras and IMUs is reduced. By combining multi-view video and IMU input, it is able to recover the full 6-DoF pose, without drift in global position. The system can operate both in constrained studio environments and in unconstrained setups such as outdoor scenes with varying illumination, moving backgrounds and occlusion. The detection sorter is able to assign 2D detections to their corresponding tracked subjects. Missing or outlier keypoint detections and even short periods of complete occlusion

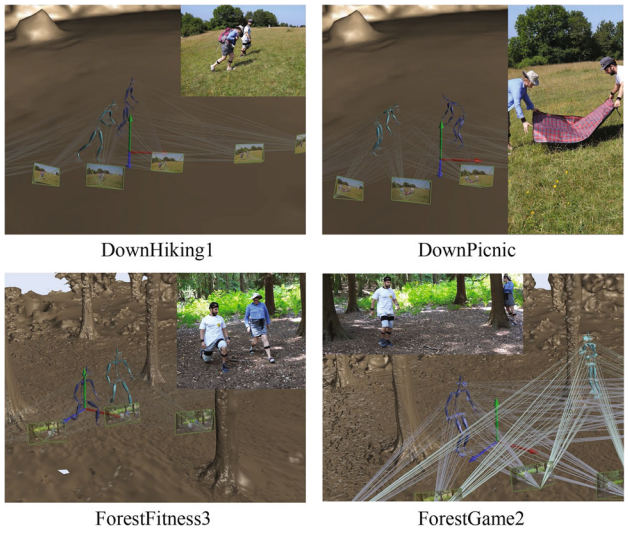


Fig. 14 Results for four sequence of the two-person outdoor dataset *Outdoor Duo* (7 cameras, 2 × 13 IMUs)

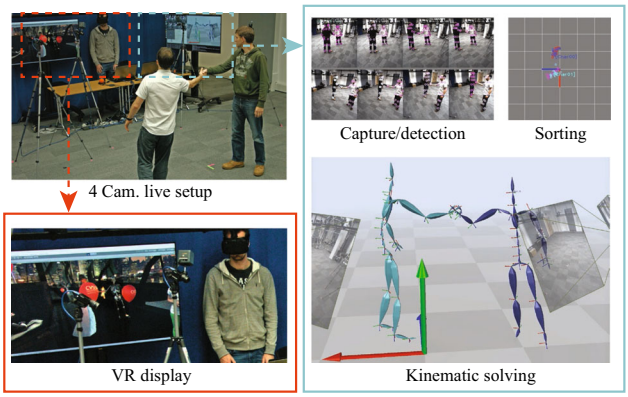


Fig. 15 Light-weight live implementation of the proposed approach with 4 camera capture and solving running on a single laptop PC (capture area approx. 3 × 3 m). The solved motion is streamed live to a game engine environment on a second laptop and displayed on a virtual reality headset

can be handled due to the robust cost function incorporating multi-modal input and a PCA-based statistical pose prior.

A limitation of our approach is that the available acceleration input from the IMUs is not being effectively utilized. Our experiments show no benefit to including the acceleration term. Using acceleration directly in a frame-by-frame solve

Table 5 Details of the recorded multi-person datasets

Dataset	Num. seqs./subjects	Num. cams.	Cam. config.	IMUs per subject
Karate	12 × 1 sub., 3 × 2 sub.	8	720p @ 30fps, sync, 360° ring	13
Ping Pong	5 × 2 sub., 1 × 4 sub.	6	720p @ 60fps, approx. sync, 120° arc	17
Outdoor Duo	17 × 2 sub.	7	720p @ 60fps, approx. sync, 120° arc	13

For all datasets, Xsens MTw wireless IMUs are used. For the *Karate* dataset, the cameras are hardware synchronised; for the *Ping Pong* and *Outdoor Duo* datasets, the cameras are approximately synchronized (to the nearest frame) using a clapper board

proves problematic due to integration error, as is noted by von Marcard et al. (2017). Future work could investigate extending the proposed approach to solve a small window of frames simultaneously thus more robustly incorporating the acceleration information, maintaining online real-time performance at the expense of a small additional latency as is done in 'deep inertial poser' (Huang et al. 2018). Another avenue for further work is improving the robustness of the detection sorter in challenging scenarios where the subjects are very close together, or moving very fast, for example by using cues from image appearance in the region of the detected keypoints. Finally, we note that to the best of our knowledge, no dataset is currently available featuring multiple-view video, IMU and optical ground truth capture of multiple subjects in uncontrolled conditions. Such a dataset, while challenging to acquire, would allow for detailed quantitative evaluation of multi-person motion capture in the wild and is therefore an interesting avenue of future investigation.

Acknowledgements This work was supported by the Innovate UK *Total Capture* project (grant 102685) and the European Union Horizon 2020 *Visual Media* project (grant 687800). We wish to thank Marco Volino, Maggie Kosek, Joao Regateiro, Lewis Bridgemam, Hansung Kim and Matt Shere for their help with data capture and Andrew Gilbert for his help with additional evaluation.

References

Agarwal, S., & Mierle, K, et al. (2017). Ceres solver. Retrieved July 20, 2017 from http://ceres-solver.org.

Alp Güler, R., Neverova, N., Kokkinos, I. (2018). Densepose: Dense human pose estimation in the wild. In *Conference on computer vision and pattern recognition (CVPR)*.

Andrews, S., Huerta, I., Komura, T., Sigal, L., & Mitchell, K. (2016). Real-time physics-based motion capture with sparse sensors. In *Proceedings of the 13th European conference on visual media production (CVMP 2016)*. https://doi.org/10.1145/2998559.2998564.

Cao, Z., Simon, T., Wei, S. E., & Sheikh, Y. (2017). Realtime multi-person 2D pose estimation using part affinity fields. In *Conference on computer vision and pattern recognition (CVPR)*.

Captury, T. (2017). The Captury markerless motion capture technology. Retrieved July 20, 2017 from http://thecaptury.com/.

Elhayek, A., De Aguiar, E., Jain, A., Tompson, J., Pishchulin, L., Andriluka, M., et al. (2015). Efficient ConvNet-based marker-less motion capture in general scenes with a low number of cameras. In *Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3810–3818). https://doi.org/10.1109/CVPR.2015.7299005.

Helten, T., Muller, M., Seidel, H. P., & Theobalt, C. (2013). Real-time body tracking with one depth camera and inertial sensors. In *Proceedings of the IEEE international conference on computer vision (ICCV)* (pp. 1105–1112).

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. In *Neural computation* (Vol. 9, pp. 1735–1780). MIT Press.

Huang, Y., Kaufmann, M., Aksan, E., Black, M. J., Hilliges, O., & Pons-Moll, G. (2018). 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, two first authors contributed equally.

Ichim, A. E., & Tombari, F. (2016). Semantic parametric body shape estimation from noisy depth sequences. *Robotics and Autonomous Systems*, *75*, 539–549. https://doi.org/10.1016/j.robot.2015.09.029.

IKinema. (2017). IKinema Orion. Retrieved July 20, 2017 from https://ikinema.com/orion.

Ionescu, C., Papava, D., Olaru, V., & Sminchisescu, C. (2014). Human3.6M: Large scale datasets and predictive methods for 3D human sensing in natural environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *36*(7), 1325–1339.

Joo, H., Simon, T., & Sheikh, Y. (2018). Total capture: A 3d deformation model for tracking faces, hands, and bodies. In *Conference on computer vision and pattern recognition (CVPR)*.

Li, S., Zhang, W., & Chan, A. B. (2017). Maximum-margin structured learning with deep networks for 3D human pose estimation. In *International conference on computer vision (ICCV)*.

Lin, M., Lin, L., Liang, X., Wang, K., & Cheng, H. (2017). Recurrent 3D pose sequence machines. In *Conference on computer vision and pattern recognition (CVPR)*.

Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., & Black, M. J. (2015). SMPL: A skinned multi-person linear model. *ACM Transactions on Graphics (Proc SIGGRAPH Asia)*, *34*(6), 248:1–248:16.

Malleson, C., Volino, M., Gilbert, A., Trumble, M., Collomosse, J., Hilton, A. (2017). Real-time full-body motion capture from video and imus. In *2017 fifth international conference on 3D vision (3DV)*.

Martinez, J., Hossain, R., Romero, J., & Little, J. J. (2017). A simple yet effective baseline for 3d human pose estimation. In *2017 IEEE international conference on computer vision (ICCV)* (pp. 2659–2668).

Mehta, D., Sotnychenko, O., Mueller, F., Xu, W., Sridhar, S., Pons-Moll, G., et al. (2018). Single-shot multi-person 3d pose estimation from monocular rgb. In *International conference on 3D vision (3DV)*.

Mehta, D., Sridhar, S., Sotnychenko, O., Rhodin, H., Shafiei, M., Seidel, H. P., et al. (2017). VNect: Real-time 3D human pose estimation with a single RGB camera. *ACM Transactions on Graphics. doi*, *10*(1145/3072959), 3073596.

OptiTrack. (2017). OptiTrack motive. Retrieved July 20, 2017 from http://www.optitrack.com.

PerceptionNeuron. (2017). Perception neuron. Retrieved July 20, 2017 from http://www.neuronmocap.com.

Rhodin, H., Richardt, C., Casas, D., Insafutdinov, E., Shafiei, M., Seidel, H. P., et al. (2016a). EgoCap: Egocentric marker-less motion capture with two fisheye cameras. *ACM Transaction on Graphics (TOG)*, *35*(6), 162:1–162:11.

Rhodin, H., Robertini, N., Casas, D., Richardt, C., Seidel, H. P., & Theobalt, C. (2016b). General automatic human shape and motion capture using volumetric contour cues. In *European conference on computer vision (ECCV)* (pp. 509–526). https://doi.org/10.1007/978-3-319-46448-0.

Roetenberg, D., Luinge, H., & Slycke, P. (2013). *Xsens MVN: Full 6DOF human motion tracking using miniature inertial sensors*. Technical report, pp. 1–7.

Rosenhahn, B., Schmaltz, C., Brox, T., Weickert, J., & Seidel, H. P. (2008). Staying well grounded in markerless motion capture. In: Pattern recognition DAGM (pp. 385–395). https://doi.org/10.1007/978-3-540-69321-5_39.

Tekin, B., Márquez-Neila, P., Salzmann, M., & Fua, P. (2016). Fusing 2D uncertainty and 3D cues for monocular body pose estimation. CoRR, arXiv:1611.05708.

Tome, D., Russell, C., Agapito, L. (2017). Lifting from the deep: Convolutional 3D pose estimation from a single image. In *Conference on computer vision and pattern recognition (CVPR)*.

Tome, D., Toso, M., Agapito, L., & Russell, C. (2018). Rethinking pose in 3d: Multi-stage refinement and recovery for markerless motion capture. In *2018 international conference on 3D vision (3DV)* (pp. 474–483). https://doi.org/10.1109/3DV.2018.00061.

Trumble, M., Gilbert, A., Hilton, A., & Collomosse, J. (2016). Deep convolutional networks for marker-less human pose estimation from multiple views. In *Proceedings of the 13th European conference on visual media production (CVMP 2016)*.

Trumble, M., Gilbert, A., Hilton, A., & Collomosse, J. (2018). Deep autoencoder for combined human pose estimation and body model upscaling. In *European conference on computer vision (ECCV)*. https://doi.org/10.1016/j.scitotenv.2003.11.003. arXiv:1807.01511.

Trumble, M., Gilbert, A., Malleson, C., Hilton, A., & Collomosse, J. (2017). Total capture: 3D human pose estimation fusing video and inertial sensors. In *British machine vision conference (BMVC)*.

Vicon. (2017). Vicon blade. Retrieved July 20, 2017 from http://www.vicon.com.

von Marcard, T., Henschel, R., Black, M., Rosenhahn, B., & Pons-Moll, G. (2018). Recovering accurate 3d human pose in the wild using imus and a moving camera. In *European conference on computer vision (ECCV)*.

Von Marcard, T., Pons-Moll, G., & Rosenhahn, B. (2016). Human pose estimation from video and IMUs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *38*(8), 1533–1547. https://doi.org/10.1109/TPAMI.2016.2522398.

von Marcard, T., Rosenhahn, B., Black, M., & Pons-Moll, G. (2017). Sparse inertial poser: Automatic 3D human pose estimation from sparse IMUs. In *Eurographics 2017* (Vol. 36).

Wei, S. E., Ramakrishna, V., Kanade, T., & Sheikh, Y. (2016). Convolutional pose machines. In *IEEE conference on computer vision and pattern recognition* (pp. 4724–4732). https://doi.org/10.1109/CVPR.2016.511, arXiv:1602.00134.

Wei, X., Zhang, P., & Chai, J. (2012). Accurate realtime full-body motion capture using a single depth camera. *ACM Transactions on Graphics*, *31*(6), 1. https://doi.org/10.1145/2366145.2366207.

Zanfir, A., Marinoiu, E., & Sminchisescu, C. (2018). Monocular 3D pose and shape estimation of multiple people in natural scenes: The importance of multiple scene constraints. In *Conference on computer vision and pattern recognition (CVPR)* (pp. 2148–2157). https://doi.org/10.1109/CVPR.2018.00229.

Zhang, Z. (1999). Flexible camera calibration by viewing a plane from unknown orientations. In *International conference on computer vision (ICCV)* (Vol. 1, pp. 666–673). https://doi.org/10.1109/ICCV.1999.791289.

Zhao, M., Li, T., Alsheikh, M. A., Tian, Y., Zhao, H., Torralba, A., et al. (2018). Through-wall human pose estimation using radio signals. In *Conference on computer vision and pattern recognition (CVPR)* (pp. 7356–7365). https://doi.org/10.1109/CVPR.2018.00768, arXiv:1011.1669v3.

Zhou, X., Zhu, M., Leonardos, S., Derpanis, K. G., & Daniilidis, K. (2016). Sparseness meets deepness: 3D human pose estimation from monocular video. In *Conference on computer vision and pattern recognition (CVPR)* (pp. 4966–4975).