PhysCap: Physically Plausible Monocular 3D Motion Capture in Real Time

SOSHI SHIMADA, Max Planck Institute for Informatics, Saarland Informatics Campus
VLADISLAV GOLYANIK, Max Planck Institute for Informatics, Saarland Informatics Campus
WEIPENG XU, Facebook Reality Labs
CHRISTIAN THEOBALT, Max Planck Institute for Informatics, Saarland Informatics Campus
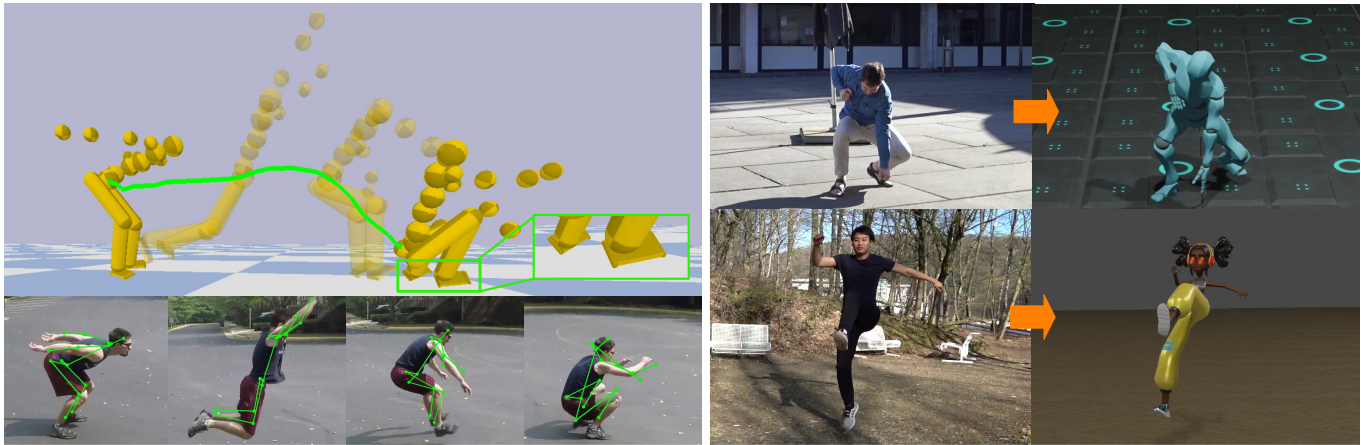
Fig. 1. *PhysCap* captures global 3D human motion in a physically plausible way from monocular videos in real time, automatically and without the use of markers. (Left:) Video of a *standing long jump* [Peng et al. 2018] and our 3D reconstructions. Thanks to its formulation on the basis of physics-based dynamics, our algorithm recovers challenging 3D human motion observed in 2D while significantly mitigating artefacts such as foot sliding, foot-floor penetration, unnatural body leaning and jitter along the depth channel that troubled earlier monocular pose estimation methods. (Right:) Since the output of *PhysCap* is environment-aware and the returned root position is global, it is directly suitable for virtual character animation, without any further post-processing. The 3D characters are taken from [Adobe 2020]. See our supplementary video for further results and visualisations.

Marker-less 3D human motion capture from a single colour camera has seen significant progress. However, it is a very challenging and severely ill-posed problem. In consequence, even the most accurate state-of-the-art approaches have significant limitations. Purely kinematic formulations on the basis of individual joints or skeletons, and the frequent frame-wise reconstruction in state-of-the-art methods greatly limit 3D accuracy and temporal stability compared to multi-view or marker-based motion capture. Further, captured 3D poses are often physically incorrect and biomechanically implausible, or exhibit implausible environment interactions (floor penetration, foot skating, unnatural body leaning and strong shifting in depth), which is problematic for any use case in computer graphics.

We, therefore, present *PhysCap*, the first algorithm for physically plausible, real-time and marker-less human 3D motion capture with a single colour camera at 25 fps. Our algorithm first captures 3D human poses purely kinematically. To this end, a CNN infers 2D and 3D joint positions, and subsequently, an inverse kinematics step finds space-time coherent joint angles and global 3D pose. Next, these kinematic reconstructions are used as constraints in a real-time physics-based pose optimiser that accounts for environment constraints (*e.g.,* collision handling and floor placement), gravity, and biophysical plausibility of human postures. Our approach employs a combination of ground reaction force and residual force for plausible root control, and uses a trained neural network to detect foot contact events in images. Our method captures physically plausible and temporally stable global 3D human motion, without physically implausible postures, floor penetrations or foot skating, from video in real time and in general scenes. *PhysCap* achieves state-of-the-art accuracy on established pose benchmarks, and we propose new metrics to demonstrate the improved physical plausibility and temporal stability.

CCS Concepts: • **Computing methodologies → Computer graphics**; **Motion capture**.

Additional Key Words and Phrases: Monocular Motion Capture, Physics-Based Constraints, Real Time, Human Body, Global 3D

ACM Reference Format:
Soshi Shimada, Vladislav Golyanik, Weipeng Xu, and Christian Theobalt. 2020. PhysCap: Physically Plausible Monocular 3D Motion Capture in Real

1 INTRODUCTION

3D human pose estimation from monocular RGB images is a very active area of research. Progress is fueled by many applications with an increasing need for reliable, real time and simple-to-use pose estimation. Here, applications in character animation, VR and AR, telepresence, or human-computer interaction, are only a few examples of high importance for graphics.

Monocular and markerless 3D capture of the human skeleton is a highly challenging and severely underconstrained problem [Kovalenko et al. 2019; Martinez et al. 2017; Mehta et al. 2017b; Pavlakos et al. 2018; Wandt and Rosenhahn 2019]. Even the best state-of-the-art algorithms, therefore, exhibit notable limitations. Most methods capture pose kinematically using individually predicted joints but do not produce smooth joint angles of a coherent kinematic skeleton. Many approaches perform per-frame pose estimates with notable temporal jitter, and reconstructions are often in root-relative but not global 3D space. Even if a global pose is predicted, depth prediction from the camera is often unstable. Also, interaction with the environment is usually entirely ignored, which leads to poses with severe collision violations, *e.g.,* floor penetration or the implausible foot sliding and incorrect foot placement. Established kinematic formulations also do not explicitly consider biomechanical plausibility of reconstructed poses, yielding reconstructed poses with improper balance, inaccurate body leaning, or temporal instability.

We note that all these artefacts are particularly problematic in the aforementioned computer graphics applications, in which temporally stable and visually plausible motion control of characters from all virtual viewpoints, in global 3D, and with respect to the physical environment, are critical. Further on, we note that established metrics in widely-used 3D pose estimation benchmarks [Ionescu et al. 2013; Mehta et al. 2017a], such as mean per joint position error (MPJPE) or 3D percentage of correct keypoints (3D-PCK), which are often even evaluated after a 3D rescaling or Procrustes alignment, do not adequately measure these artefacts. In fact, we show (see Sec. 4, and supplemental video) that even some top-performing methods on these benchmarks produce results with substantial temporal noise and unstable depth prediction, with frequent violation of environment constraints, and with frequent disregard of physical and anatomical pose plausibility. In consequence, there is still a notable gap between monocular 3D pose human estimation approaches and the gold standard accuracy and motion quality of suit-based or marker-based motion capture systems, which are unfortunately expensive, complex to use and not suited for many of the aforementioned applications requiring in-the-wild capture.

We, therefore, present *PhysCap* – a new approach for easy-to-use monocular global 3D human motion capture that significantly narrows this gap and substantially reduces the aforementioned artefacts, see Fig. 1 for an overview. *PhysCap* is, to our knowledge, the first method that jointly possesses all the following properties: it is fully-automatic, markerless, works in general scenes, runs in real time, captures a space-time coherent skeleton pose and global 3D pose sequence of state-of-the-art temporal stability and smoothness. It exhibits state-of-the-art posture and position accuracy, and

captures physically and anatomically plausible poses that correctly adhere to physics and environment constraints. To this end, we rethink and bring together in new way ideas from kinematics-based monocular pose estimation and physics-based human character animation.

The *first stage* of our algorithm is similar to [Mehta et al. 2017b] and estimates 3D body poses in a purely kinematic, physics-agnostic way. A convolutional neural network (CNN) infers combined 2D and 3D joint positions from an input video, which are then refined in a space-time inverse kinematics to yield the first estimate of skeletal joint angles and global 3D poses. In the *second stage*, the foot contact and the motion states are predicted for every frame. Therefore, we employ a new CNN that detects heel and forefoot placement on the ground from estimated 2D keypoints in images, and classifies the observed poses into stationary or non-stationary. In the *third stage*, the final physically plausible 3D skeletal joint angle and pose sequence is computed in real time. This stage regularises human motion with a torque-controlled physics-based character represented by a kinematic chain with a floating base. To this end, the optimal control forces for each degree of freedom (DoF) of the kinematic chain are computed, such that the kinematic pose estimates from the first stage – in both 2D and 3D – are reproduced as closely as possible. The optimisation ensures that physics constraints like gravity, collisions, foot placement, as well as physical pose plausibility (*e.g.,* balancing), are fulfilled. To summarise, our **contributions** in this article are:

- The first, to the best of our knowledge, marker-less monocular 3D human motion capture approach on the basis of an explicit physics-based dynamics model which runs in real time and captures global, physically plausible skeletal motion (Sec. 4).
- A CNN to detect foot contact and motion states from images (Sec. 4.2).
- A new pose optimisation framework with a human parametrised by a torque-controlled simulated character with a floating base and PD joint controllers; it reproduces kinematically captured 2D/3D poses and simultaneously accounts for physics constraints like ground reaction forces, foot contact states and collision response (Sec. 4.3).
- Quantitative metrics to assess frame-to-frame jitter and floor penetration in captured motions (Sec. 5.3.1).
- Physically-justified results with significantly fewer artefacts, such as frame-to-frame jitter, incorrect leaning, foot sliding and floor penetration than related methods (confirmed by a user study and metrics), as well as state-of-the-art 2D and 3D accuracy and temporal stability (Sec. 5).

We demonstrate the benefits of our approach through experimental evaluation on several datasets (including newly recorded videos) against multiple state-of-the-art methods for monocular 3D human motion capture and pose estimation.

2 RELATED WORK

Our method mainly relates to two different categories of approaches – (markerless) 3D human motion capture from colour imagery, and physics-based character animation. In the following, we review related types of methods, focusing on the most closely related works.

Multi-View Methods for 3D Human Motion Capture from RGB. Reconstructing humans from multi-view images is well studied. Multi-view motion capture methods track the articulated skeletal motion, usually by fitting an articulated template to imagery [Bo and Sminchisescu 2010; Brox et al. 2010; Elhayek et al. 2016, 2014; Gall et al. 2010; Stoll et al. 2011; Wang et al. 2018; Zhang et al. 2020].

Other methods, sometimes termed performance capture methods, additionally capture the non-rigid surface deformation, *e.g.,* of clothing [Cagniart et al. 2010; Starck and Hilton 2007; Vlasic et al. 2009; Waschbüsch et al. 2005]. They usually fit some form of a template model to multi-view imagery [Bradley et al. 2008; De Aguiar et al. 2008; Martin-Brualla et al. 2018] that often also has an underlying kinematic skeleton [Gall et al. 2009; Liu et al. 2011; Vlasic et al. 2008; Wu et al. 2012]. Multi-view methods have demonstrated compelling results and some enable free-viewpoint video. However, they require expensive multi-camera setups and often controlled studio environments.

Monocular 3D Human Motion Capture and Pose Estimation from RGB. Marker-less 3D human pose estimation (reconstruction of 3D joint positions only) and motion capture (reconstruction of global 3D body motion and joint angles of a coherent skeleton) from a single colour or greyscale image are highly ill-posed problems. The state of the art on monocular 3D human pose estimation has greatly progressed in recent years, mostly fueled by the power of trained CNNs [Habibie et al. 2019; Mehta et al. 2017a]. Some methods estimate 3D pose by combining 2D keypoints prediction with body depth regression [Dabral et al. 2018; Newell et al. 2016; Yang et al. 2018; Zhou et al. 2017] or with regression of 3D joint location probabilities [Mehta et al. 2017b; Pavlakos et al. 2017] in a trained CNN. Lifting methods predict joint depths from detected 2D keypoints [Chen and Ramanan 2017; Martinez et al. 2017; Pavlakos et al. 2018; Tomè et al. 2017]. Other CNNs regress 3D joint locations directly [Mehta et al. 2017a; Rhodin et al. 2018; Tekin et al. 2016]. Another category of methods combines CNN-based keypoint detection with constraints from a parametric body model, *e.g.*, by using reprojection losses during training [Bogo et al. 2016; Brau and Jiang 2016; Habibie et al. 2019]. Some works approach monocular multi-person 3D pose estimation [Rogez et al. 2019] and motion capture [Mehta et al. 2020], or estimate non-rigidly deforming human surface geometry from monocular video on top of skeletal motion [Habermann et al. 2020, 2019; Xu et al. 2020]. In addition to greyscale images, [Xu et al. 2020] use an asynchronous event stream from an event camera as input. Both these latter directions are complementary but orthogonal to our work.

The majority of methods in this domain estimates 3D pose as a root-relative 3D position of the body joints [Kovalenko et al. 2019; Martinez et al. 2017; Moreno-Noguer 2017; Pavlakos et al. 2018; Wandt and Rosenhahn 2019]. This is problematic for applications in graphics, as temporal jitter, varying bone lengths and the often not recovered global 3D pose make animating virtual characters hard. Other monocular methods are trained to estimate parameters or joint angles of a skeleton [Zhou et al. 2016] or parametric model [Kanazawa et al. 2018]. [Mehta et al. 2020, 2017b] employ inverse kinematics on top of CNN-based 2D/3D inference to obtain joint angles of a coherent skeleton in global 3D and in real-time.

Results of all aforementioned methods frequently violate laws of physics, and exhibit foot-floor penetrations, foot sliding, and unbalanced or implausible poses floating in the air, as well as notable jitter. Some methods try to reduce jitter by exploiting temporal information [Kanazawa et al. 2019; Kocabas et al. 2020], *e.g.,* by estimating smooth multi-frame scene trajectories [Peng et al. 2018]. [Zou et al. 2020] try to reduce foot sliding by ground contact constraints. [Zanfir et al. 2018] jointly reason about ground planes and volumetric occupancy for multi-person pose estimation. [Monszpart et al. 2019] jointly infer coarse scene layout and human pose from monocular interaction video, and [Hassan et al. 2019] use a pre-scanned 3D model of scene geometry to constrain kinematic pose optimisation. To overcome the aforementioned limitations, no prior work formulates monocular motion capture on the basis of an explicit physics-based dynamics model and in real-time, as we do.

Physics-Based Character Animation. Character animation on the basis of physics-based controllers has been investigated for many years [Barzel et al. 1996; Sharon and van de Panne 2005; Wrotek et al. 2006], and remains an active area of research, [Andrews et al. 2016; Bergamin et al. 2019; Levine and Popović 2012; Zheng and Yamane 2013]. [Levine and Popović 2012] employ a quasi-physical simulation that approximates a reference motion trajectory in real-time. They can follow non-physical reference motion by applying a direct actuation at the root. By using proportional derivative (PD) controllers and computing optimal torques and contact forces, [Zheng and Yamane 2013] make a character follow a reference motion captured while keeping balance. [Liu et al. 2010] proposed a probabilistic algorithm for physics-based character animation. Due to the stochastic property and inherent randomness, their results evince variations, but the method requires multiple minutes of runtime per sequence. Andrews *et al.* [2016] employ rigid dynamics to drive a virtual character from a combination of marker-based motion capture and body-mounted sensors. This animation setting is related to motion transfer onto robots. [Nakaoka et al. 2007] transferred human motion captured by a multi-camera marker-based system onto a robot, with an emphasis on leg motion. [Zhang et al. 2014] leverage depth cameras and wearable pressure sensors and apply physics-based motion optimisation. We take inspiration from these works for our setting, where we have to capture in a physically correct way and in real time global 3D human motion from images, using intermediate pose reconstruction results that exhibit notable artefacts and violations of physics laws. *PhysCap*, therefore, combines an initial kinematics-based pose reconstruction with PD controller based physical pose optimisation.

Several recent methods apply deep reinforcement learning to virtual character animation control [Bergamin et al. 2019; Lee et al. 2019; Peng et al. 2018]. Peng *et al.* [2018] propose a reinforcement learning approach for transferring dynamic human performances observed in monocular videos. They first estimate smooth motion trajectories with recent monocular human pose estimation techniques, and then train an imitating control policy for a virtual character. [Bergamin et al. 2019] train a controller for a virtual character from several minutes of motion capture data which covers the expected variety of motions and poses. Once trained, the virtual character can follow directional commands of the user in real time,

while being robust to collisional obstacles. Other work [Lee et al. 2019] combines a muscle actuation model with deep reinforcement learning. [Jiang et al. 2019] express an animation objective in muscle actuation space. The work on learning animation controllers for specific motion classes is inspirational but different from real-time physics-based motion capture of general motion.

Physically Plausible Monocular 3D Human Motion Capture. Only a few works on monocular 3D human motion capture using explicit physics-based constraints exist [Li et al. 2019; Vondrak et al. 2012; Wei and Chai 2010; Zell et al. 2017]. [Wei and Chai 2010] capture 3D human poses from uncalibrated monocular video using physics constraints. Their approach requires manual user input for each frame of a video. In contrast, our approach is automatic, runs in real time, and uses a different formulation for physics-based pose optimisation geared to our setting. [Vondrak et al. 2012] capture bipedal controllers from a video. Their controllers are robust to perturbations and generalise well for a variety of motions. However, unlike our *PhysCap*, the generated motion often looks unnatural and their method does not run in real time. [Zell et al. 2017] capture poses and internal body forces from images only for certain classes of motion (*e.g.,* lifting and walking) by using a data-driven approach, but not an explicit forward dynamics approach handling a wide range of motions, like ours.

Our *PhysCap* bears most similarities with the rigid body dynamics based monocular human pose estimation by Li *et al.* [2019]. Li *et al.* estimate 3D poses, contact states and forces from input videos with physics-based constraints. However, their method and our approach are substantially different. While Li *et al.* focus on object-person interactions, we target a variety of general motions, including complex acrobatic motions such as backflipping without objects. Their method does not run in real time and requires manual annotations on images to train the contact state estimation networks. In contrast, we leverage the PD controller based inverse dynamics tracking, which results in physically plausible, smooth and natural skeletal pose and root motion capture in real time. Moreover, our contact state estimation network relies on annotations generated in a semi-automatic way. This enables our architecture to be trained on large datasets, which results in the improved generalisability. No previous method of the reviewed category "physically plausible monocular 3D human motion capture" combines *the ability of our algorithm to capture global 3D human pose of similar quality and physical plausibility in real time.*

3 BODY MODEL AND PRELIMINARIES

The input to *PhysCap* is a 2D image sequence $\mathbf{I}_t$, $t \in \{1, \dots, T\}$, where T is the total number of frames and t is the frame index. We assume a perspective camera model and calibrate the camera and floor location before tracking starts. Our approach outputs a physically plausible real-time 3D motion capture result $\mathbf{q}^t_{phys} \in \mathbb{R}^m$ (where m is the number of degrees of freedom) that adheres to the image observation, as well as physics-based posture and environment constraints. For our human model, $m = 43$. Joint angles are parametrised by Euler angles. The mass distribution of our character is computed following [Liu et al. 2010]. Our character model has a skeleton composed of 37 joints and *links*. A link defines the
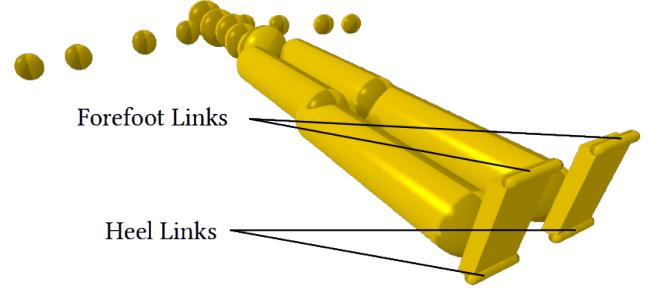
Fig. 2. Our virtual character used in stage III. The forefoot and heel links are involved in the mesh collision checks with the floor plane in the physics engine [Coumans and Bai 2016].

volumetric extent of a body part via a collision proxy. The forefoot and heel links, centred at the respective joints of our character (see Fig. 2), are used to detect foot-floor collisions during physics-based pose optimisation.

Throughout our algorithm, we represent the pose of our character by a combined vector $\mathbf{q} \in \mathbb{R}^m$ [Featherstone 2014]. The first three entries of $\mathbf{q}$ contain the global 3D root position in Cartesian coordinates, the next three entries encode the orientation of the root, and the remaining entries are the joint angles. When solving for the physics-based motion capture result, the motion of the physics-based character will be controlled by the vector of forces denoted by $\boldsymbol{\tau} \in \mathbb{R}^m$ interacting with gravity, Coriolis and centripetal forces $\mathbf{c} \in \mathbb{R}^m$. The root of our character is not fixed and can globally move in the environment, which is commonly called a floating-base system. Let the velocity and acceleration of $\mathbf{q}$ be $\dot{\mathbf{q}} \in \mathbb{R}^m$ and $\ddot{\mathbf{q}} \in \mathbb{R}^m$, respectively. Using the finite-difference method, the relationship between $\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$ can be written as

$$\begin{aligned} \dot{\mathbf{q}}^{i+1} &= \dot{\mathbf{q}}^i + \phi\ddot{\mathbf{q}}^i, \\ \mathbf{q}^{i+1} &= \mathbf{q}^i + \phi\dot{\mathbf{q}}^{i+1}, \end{aligned} \quad (1)$$

where i represents the simulation step index and $\phi = 0.01$ is the simulation step size.

For the motion to be physically plausible, $\ddot{\mathbf{q}}$ and the vector of forces $\boldsymbol{\tau}$ must satisfy the equation of motion [Featherstone 2014]:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} - \boldsymbol{\tau} = \mathbf{J}^T\mathbf{G}\lambda - \mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}), \quad (2)$$

where $\mathbf{M} \in \mathbb{R}^{m \times m}$ is a joint space inertia matrix which is composed of the moment of inertia of the system. It is computed using the Composite Rigid Body algorithm [Featherstone 2014]. $\mathbf{J} \in \mathbb{R}^{6N_c \times m}$ is a contact Jacobi matrix which relates the external forces to joint coordinates, with N_c denoting the number of links where the contact force is applied. $\mathbf{G} \in \mathcal{R}^{6N_c \times 3N_c}$ transforms contact forces $\lambda \in \mathcal{R}^{3N_c}$ into the linear force and torque [Zheng and Yamane 2013].

Usually, in a floating-base system, the first six entries of $\boldsymbol{\tau}$ which correspond to the root motion are set to 0 for a humanoid character control. This reflects the fact that humans do not directly control root translation and orientation by muscles acting on the root, but indirectly by the other joints and muscles in the body. In our case, however, the kinematic pose $\mathbf{q}^t_{kin}$ which our final physically plausible result shall reproduce as much as possible (see Sec. 4),
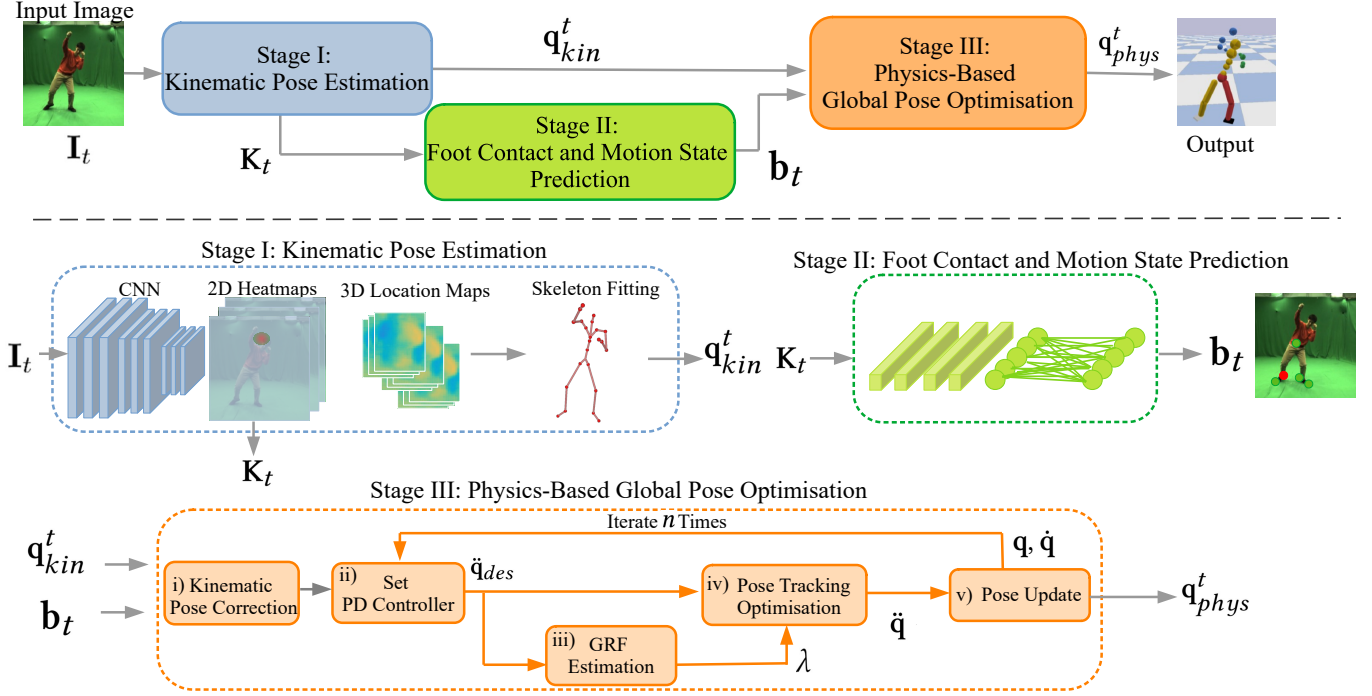
Fig. 3. **Overview of our pipeline.** In stage I, the 3D pose estimation network accepts RGB image $\mathbf{I}_t$ as input and returns 2D joint keypoints $\mathbf{K}_t$ along with the global 3D pose $\mathbf{q}_{kin}^t$, *i.e.*, root translation, orientation and joint angles of a kinematic skeleton. In stage II, $\mathbf{K}_t$ is fed to the contact and motion state detection network. Stage II returns the contact states of heels and forefeet as well as a label $\mathbf{b}_t$ that represents if the subject in $\mathbf{I}_t$ is stationary or not. In stage III, $\mathbf{q}_{kin}^t$ and $\mathbf{b}_t$ are used to iteratively update the character pose respecting physics laws. After the n pose update iterations, we obtain the final 3D pose $\mathbf{q}_{phys}^t$. Note that the orange arrows in stage III represent the steps that are repeated in the loop in every iteration. Kinematic pose correction is performed only once at the beginning of stage III.

is estimated from a monocular image sequence (see stage I in Fig. 3), which contains physically implausible artefacts. Solving for joint torque controls that blindly make the character follow, would make the character quickly fall down. Hence, we keep the first six entries of $\boldsymbol{\tau}$ in our formulation and can thus directly control the root position and orientation with an additional external force. This enables the final character motion to keep up with the global root trajectory estimated in the first stage of *PhysCap*, without falling down.

4 METHOD

Our *PhysCap* approach includes three stages, see Fig. 3 for an overview. The first stage performs *kinematic pose estimation*. This encompasses 2D heatmap and 3D location map regression for each body joint with a CNN, followed by a model-based space-time pose optimisation step (Sec. 4.1). This stage returns 3D skeleton pose in joint angles $\mathbf{q}_{kin}^t \in \mathbb{R}^m$ along with the 2D joint keypoints $\mathbf{K}_t \in \mathbb{R}^{s \times 2}$ for every image; s denotes the number of 2D joint keypoints. As explained earlier, this initial kinematic reconstruction $\mathbf{q}_{kin}^t$ is prone to physically implausible effects such as foot-floor penetration, foot skating, anatomically implausible body leaning and temporal jitter, especially notable along the depth dimension.

The second stage performs *foot contact and motion state detection*, which uses 2D joint detections $\mathbf{K}_t$ to classify the poses reconstructed

so far into stationary and non-stationary – this is stored in one binary flag. It also estimates binary foot-floor contact flags, *i.e.,* for the toes and heels of both feet, resulting in four binary flags (Sec. 4.2). This stage outputs the combined state vector $\mathbf{b}_t \in \mathbb{R}^5$.

The third and final stage of *PhysCap* is the *physically plausible global 3D pose estimation* (Sec. 4.3). It combines the estimates from the first two stages with physics-based constraints to yield a physically plausible real-time 3D motion capture result that adheres to physics-based posture and environment constraints $\mathbf{q}_{phys}^t \in \mathbb{R}^m$. In the following, we describe each of the stages in detail.

4.1 Stage I: Kinematic Pose Estimation

Our kinematic pose estimation stage follows the real-time VNect algorithm [Mehta et al. 2017b], see Fig. 3, stage I. We first predict heatmaps of 2D joints and root-relative location maps of joint positions in 3D with a specially tailored fully convolutional neural network using a ResNet [He et al. 2016] core. The ground truth joint locations for training are taken from the MPII [Andriluka et al. 2014] and LSP [Johnson and Everingham 2011] datasets in the 2D case, and MPI-INF-3DHP [Mehta et al. 2017a] and Human3.6m [Ionescu et al. 2013] datasets in the 3D case.

Next, the estimated 2D and 3D joint locations are temporally filtered and used as constraints in a kinematic skeleton fitting step
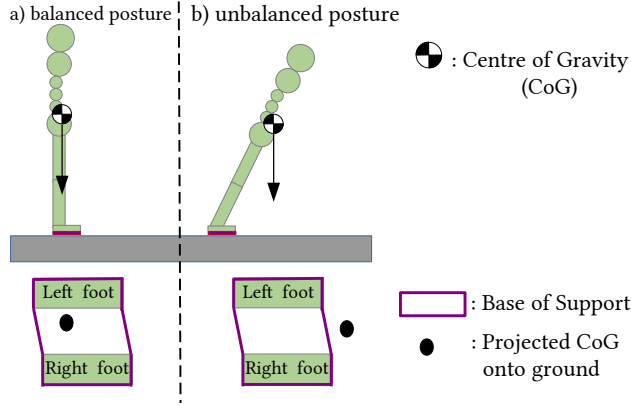
Fig. 4. (a) Balanced posture: the CoG of the body projects inside the base of support. (b) Unbalanced posture: the CoG does not project inside the base of support, which causes the human to start losing a balance.

that optimises the following energy function:

$$\mathbf{E}_{kin}(\mathbf{q}_{kin}^t) = \mathbf{E}_{IK}(\mathbf{q}_{kin}^t) + \mathbf{E}_{proj.}(\mathbf{q}_{kin}^t) + \\ \mathbf{E}_{smooth}(\mathbf{q}_{kin}^t) + \mathbf{E}_{depth}(\mathbf{q}_{kin}^t). \quad (3)$$

The energy function (3) contains four terms (see [Mehta et al. 2017b]), *i.e.*, the 3D inverse kinematics term $\mathbf{E}_{IK}$, the projection term $\mathbf{E}_{proj.}$, the temporal stability term $\mathbf{E}_{smooth}$ and the depth uncertainty correction term $\mathbf{E}_{depth}$. $\mathbf{E}_{IK}$ is the data term which constrains the 3D pose to be close to the 3D joint predictions from the CNN. $\mathbf{E}_{proj.}$ enforces the pose $\mathbf{q}_{kin}^t$ to reproject it to the 2D keypoints (joints) detected by the CNN. Note that this reprojection constraint, together with calibrated camera and calibrated bone lengths, enables computation of the global 3D root (pelvis) position in the camera space. Temporal stability is further imposed by penalising the root's acceleration and variations along the depth channel by $\mathbf{E}_{smooth}$ and $\mathbf{E}_{depth}$, respectively. The energy (3) is optimised by non-linear least squares (Levenberg-Marquardt algorithm [Levenberg 1944; Marquardt 1963]), and the obtained vector of joint angles and the root rotation and position $\mathbf{q}_{kin}^t$ of a skeleton with fixed bone lengths are smoothed by an adaptive first-order low-pass filter [Casiez et al. 2012]. Skeleton bone lengths of a human can be computed, up to a global scale, from averaged 3D joint detections of a few initial frames. Knowing the metric height of the human determines the scale factor to compute metrically correct global 3D poses.

The result of stage I is a temporally-consistent joint angle sequence but, as noted earlier, captured poses can exhibit artefacts and contradict physical plausibility (*e.g.,* evince floor penetration, incorrect body leaning, temporal jitter, *etc.*).

4.2 Stage II: Foot Contact and Motion State Detection

The ground reaction force (GRF) – applied when the feet touch the ground – enables humans to walk and control their posture. The interplay of internal body forces and the ground reaction force controls human pose, which enables locomotion and body balancing by controlling the centre of gravity (CoG). To compute physically plausible poses accounting for the GRF in stage III, we thus need
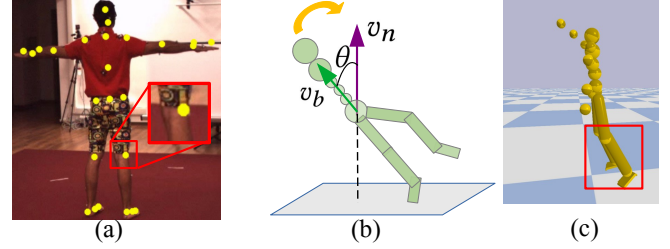
Fig. 5. (a) An exemplary frame from the Human 3.6M dataset with the ground truth reprojections of the 3D joint keypoints. The magnified view in the red rectangle shows the reprojected keypoint that deviates from the rotation centre (the middle of the knee). (b) Schematic visualisation of the reference motion correction. Readers are referred to Sec. 4.3.1 for its details. (c) Example of a visually unnatural standing (stationary) pose caused by physically implausible knee bending.

to know foot-floor contact states. Another important aspect of the physical plausibility of biped poses, in general, is balance. When a human is standing or in a stationary upright state, the CoG of her body projects inside a base of support (BoS). The BoS is an area on the ground bounded by the foot contact points, see Fig. 4 for a visualisation. When the CoG projects outside the BoS in a stationary pose, a human starts losing balance and will fall if no correcting motion or step is applied. Therefore, maintaining a static pose with an extensive leaning, as often observed in the results of monocular pose estimation, is not physically plausible (Fig .4-(b)). The aforementioned CoG projection criterion can be used to correct imbalanced stationary poses [Coros et al. 2010; Faloutsos et al. 2001; Macchietto et al. 2009]. To perform such correction in stage III, we need to know if a pose is stationary or non-stationary (whether it is a part of a locomotion/walking phase).

Stage II, therefore, estimates foot-floor contact states of the feet in each frame and determines whether the pose of the subject in $\mathbf{I}_t$ is stationary or not. To predict both, *i.e.,* foot contact and motion states, we use a neural network whose architecture extends Zou *et al.* [2020] who only predict foot contacts. It is composed of temporal convolutional layers with one fully connected layer at the end. The network takes as input all 2D keypoints $\mathbf{K}_t$ from the last seven time steps (the temporal window size is set to seven), and returns for each image frame binary labels indicating whether the subject is in the stationary or non-stationary pose, as well as the contact state flags for the forefeet and heels of both feet encompassed in $\mathbf{b}_t$. The supervisory labels for training this network are automatically computed on a subset of the 3D motion sequences of the Human3.6M [Ionescu et al. 2013] and DeepCap [Habermann et al. 2020] datasets using the following criteria: the forefoot and heel joint contact labels are computed based on the assumption that a joint in contact is not sliding, *i.e.,* the velocity is lower than 5 cm/sec. In addition, we use a height criterion, *i.e.,* the forefoot/heel, when in contact with the floor, has to be at a 3D height that is lower than a threshold $h_{thres.}$. To determine this threshold for each sequence, we calculate the average heel h_{avg}^{heel} and forefoot h_{avg}^{ffoot} heights for each subject using the first ten frames (when both feet touch the ground). Thresholds are then computed as $h_{thres.}^{heel} = h_{avg}^{heel} + 5cm$ for heels and $h_{thres.}^{ffoot} =$

h_{avg}^{ffoot} + 5cm for the forefeet. This second criterion is needed since, otherwise, a foot in the air that is kept static could also be labelled as being in contact.

We also automatically label stationary and non-stationary poses on the same sequences. When standing and walking, the CoG of the human body typically lies close to the pelvis in 3D, which corresponds to the skeletal root position in both the Human3.6M and DeepCap datasets. Therefore, when the velocity of 3D root is lower than a threshold φ_v, we classify the pose as *stationary*, and *non-stationary* otherwise. In total, around $600k$ sets of contact and motion state labels for the human images are generated.

4.3 Stage III: Physically Plausible Global 3D Pose Estimation

Stage III uses the results of stages I and II as inputs, *i.e.,* $\mathbf{q}_{kin}^t$ and $\mathbf{b}_t$. It transforms the kinematic motion estimate into a physically plausible global 3D pose sequence that corresponds to the images and adheres to anatomy and environmental constraints imposed by the laws of physics. To this end, we represent the human as a torque-controlled simulated character with a floating base and PD joint controllers [A. Salem and Aly 2015]. The core is to solve an energy-based optimisation problem to find the vector of forces τ and accelerations $\ddot{\mathbf{q}}$ of the character such that the equations of motion with constraints are fulfilled (Sec. 4.3.5). This optimisation is preceded by several preprocessing steps applied to each frame.

First i), we correct $\mathbf{q}_{kin}^t$ if it is strongly implausible based on several easy-to-test criteria (Sec. 4.3.1). Second ii), we estimate the desired acceleration $\ddot{\mathbf{q}}_{des} \in \mathbb{R}^m$ necessary to reproduce $\mathbf{q}_{kin}^t$ based on the PD control rule (Secs. 4.3.2). Third iii), in input frames in which a foot is in contact with the floor (Sec. 4.3.3), we estimate the ground reaction force (GRF) λ (Sec. 4.3.4). Fourth iv), we solve the optimisation problem (10) to estimate τ and accelerations $\ddot{\mathbf{q}}$ where the equation of motion with the estimated GRF λ and the contact constraint to avoid foot-floor penetration (Sec. 4.3.5) are integrated as constraints. Note that the contact constraint is integrated only when the foot is in contact with the floor. Otherwise, only the equation of motion without GRF is introduced as a constraint in (10). v) Lastly, the pose is updated using the finite-difference method (Eq. (1)) with the estimated acceleration $\ddot{\mathbf{q}}$. The steps ii) - v) are iterated $n = 4$ times for each frame of video.

As also observed by [Andrews et al. 2016], this two-step optimisation iii) and iv) reduces direct actuation of the character's root as much as possible (which could otherwise lead to slightly unnatural locomotion), and explains the kinematically estimated root position and orientation by torques applied to other joints as much as possible when there is a foot-floor contact. Moreover, this two-step optimisation is computationally less expensive rather than estimating $\ddot{\mathbf{q}}$, τ and λ simultaneously [Zheng and Yamane 2013]. Our algorithm thus finds a plausible balance between pose accuracy, physical accuracy, the naturalness of captured motion and real-time performance.

4.3.1 Pose Correction. Due to the error accumulation in stage I (*e.g.,* as a result of the deviation of 3D annotations from the joint rotation centres in the skeleton model, see Fig. 5-(a), as well as inaccuracies in the neural network predictions and skeleton fitting), the

estimated 3D pose $\mathbf{q}_{kin}^t$ is often not physically plausible. Therefore, prior to torque-based optimisation, we pre-correct a pose $\mathbf{q}_{kin}^t$ from stage I if it is 1) stationary and 2) unbalanced, *i.e.,* the CoG projects outside the BoS. If both correction criteria are fulfilled, we compute the angle θ_t between the ground plane normal v_n and the vector v_b that defines the direction of the spine relative to the root in the local character's coordinate system (see Fig. 5-(b) for the schematic visualisation). We then correct the orientation of the virtual character towards a posture, for which CoG projects inside BoS. Correcting θ_t in one large step could lead to instabilities in physics-based pose optimisation. Instead, we reduce θ_t by a small rotation of the virtual character around its horizontal axis (*i.e.,* the axis passing through the transverse plane of a human body) starting with the corrective angle $\xi_t = \frac{\theta_t}{10}$ for the first frame. Thereby, we accumulate the degree of correction in ξ for the subsequent frames, *i.e.,* $\xi_{t+1} = \xi_t + \frac{\theta_t}{10}$. Note that θ_t is decreasing for every frame and the correction step is performed for all subsequent frames until 1) the pose becomes non-stationary or 2) CoG projects inside BoS[1].

However, simply correcting the spine orientation by the skeleton rotation around the horizontal axis can lead to implausible standing poses, since the knees can still be unnaturally bent for the obtained upright posture (see Fig. 5-(c) for an example). To account for that, we adjust the respective DoFs of the knees and hips such that the relative orientation between upper legs and spine, as well as upper and lower legs, are more straight. The hip and knee correction starts if both correction criteria are *still* fulfilled and θ_t is *already* very small. Similarly to the θ correction, we introduce accumulator variables for every knee and every hip. The correction step for knees and hips is likewise performed until 1) the pose becomes non-stationary or 2) CoG projects inside BoS[1].

4.3.2 Computing the Desired Accelerations. To control the physics-based virtual character such that it reproduces the kinematic estimate $\mathbf{q}_{kin}^t$, we set the desired joint acceleration $\ddot{\mathbf{q}}_{des}$ following the PD controller rule:

$$\ddot{\mathbf{q}}_{des} = \ddot{\mathbf{q}}_{kin}^t + k_p(\mathbf{q}_{kin}^t - \mathbf{q}) + k_d(\dot{\mathbf{q}}_{kin}^t - \dot{\mathbf{q}}). \qquad (4)$$

The desired acceleration $\ddot{\mathbf{q}}_{des}$ is later used in the GRF estimation step (Sec. 4.3.4) and the final pose optimisation (Sec. 4.3.5). Controlling the character motion on the basis of a PD controller in the system enables the character to exert torques τ which reproduce the kinematic estimate $\mathbf{q}_{kin}^t$ while significantly mitigating undesired effects such as joint and base position jitter.

4.3.3 Foot-Floor Collision Detection. To avoid foot-floor penetration in the final pose sequence and to mitigate contact position sliding, we integrate hard constraints in the physics-based pose optimisation to enforce zero velocity of forefoot and heel links in Sec. 4.3.5. However, these constraints can lead to unnatural motion in rare cases when the state prediction network may fail to estimate the correct foot contact states (*e.g.,* when the foot suddenly stops in the air while walking). We thus update the contact state output of the state prediction network $\mathbf{b}_{t,j \in \{1,...,4\}}$, to yield $\mathbf{b}'_{t,j \in \{1,...,4\}}$ as

---

[1]either after the correction or already in $\mathbf{q}_{kin}^t$ provided by stage I

follows:

$$\mathbf{b}'_{t,j\in\{1,\dots,4\}} = \begin{cases} 1, & \text{if } (\mathbf{b}^j = 1 \text{ and } h^j < \psi) \text{ or} \\ & \text{the } j\text{-th link collides with the floor plane,} \\ 0, & \text{otherwise.} \end{cases} \quad (5)$$

This means we consider a forefoot or heel link to be in contact only if its height h^j is less than a threshold $\psi = 0.1$m above the calibrated ground plane.

In addition, we employ the *Pybullet* [Coumans and Bai 2016] physics engine to detect foot-floor collision for the left and right foot links. Note that combining the mesh collision information with the predictions from the state prediction network is necessary because 1) the foot may not touch the floor plane in the simulation when the subject's foot is actually in contact with the floor due to the inaccuracy of $\mathbf{q}^t_{kin}$, and 2) the foot can penetrate into the mesh floor plane if the network misdetects the contact state when there is actually a foot contact in $\mathbf{I}_t$.

4.3.4 Ground Reaction Force (GRF) Estimation. We first compute the GRF λ – when there is a contact between a foot and floor – which best explains the motion of the root as coming from stage I. However, the target trajectory from stage I can be physically implausible, and we will thus eventually also require a *residual* force directly applied on the root to explain the target trajectory; this force will be computed in the final optimisation. To compute the GRF, we solve the following minimisation problem:

$$\min_{\lambda} \|\mathbf{M}_1 \ddot{\mathbf{q}}_{des} - \mathbf{J}_1^T \mathbf{G}\lambda\|, \\ \text{s.t. } \lambda \in F, \quad (6)$$

where $\|\cdot\|$ denotes ℓ^2-norm, and $\mathbf{M}_1 \in \mathcal{R}^{6\times m}$ together with $\mathbf{J}_1^T \in \mathcal{R}^{6\times 6N_c}$ are the first six rows of $\mathbf{M}$ and $\mathbf{J}^T$ that correspond to the root joint, respectively. Since we do not consider sliding contact, the contact force λ has to satisfy friction cone constraints. Thus, we formulate a linearised friction cone constraint F. That is,

$$F^j = \left\{\lambda^j \in \mathcal{R}^3 | \lambda_n^j > 0, \left|\lambda_t^j\right| \leq \bar{\mu}\lambda_n^j, \left|\lambda_b^j\right| \leq \bar{\mu}\lambda_n^j\right\}, \quad (7)$$

where λ_n^j is a normal component, λ_t^j and λ_b^j are the tangential components of a contact force at the j-th contact position; μ is a friction coefficient which we set to 0.8 and the friction coefficient of inner linear cone approximation reads $\bar{\mu} = \mu/\sqrt{2}$.

The GRF λ is then integrated into the subsequent optimisation step (10) to estimate torques and accelerations of all joints in the body, including an additional residual direct root actuation component that is needed to explain the difference between the global 3D root trajectory of the kinematic estimate and the final physically correct result. The aim is to keep this direct root actuation as small as possible, which is best achieved by a two-stage strategy that first estimates the GRF separately. Moreover, we observed this two-step optimisation enables faster computation than estimating λ, $\ddot{\mathbf{q}}$ and τ all at once. It is hence more suitable for our approach which aims at real-time operation.

4.3.5 Physics-Based Pose Optimisation. In this step, we solve an optimisation problem to estimate τ and $\ddot{\mathbf{q}}$ to track $\mathbf{q}^t_{kin}$ using the equation of motion (2) as a constraint. When contact is detected (Sec. 4.3.3), we integrate the estimated ground reaction force λ

Table 1. Names and duration of our six newly recorded outdoor sequences captured using SONY DSC-RX0 at 25 fps.

Sequence ID	Sequence Name	Duration [sec]
1	*building 1*	132
2	*building 2*	90
3	*forest*	105
4	*backyard*	60
5	*balance beam 1*	21
6	*balance beam 2*	12

(Sec. 4.3.4) in the equation of motion. In addition, we introduce contact constraints to prevent foot-floor penetration and foot sliding when contacts are detected.

Let $\dot{\mathbf{r}}_j$ be the velocity of the j-th contact link. Then, using the relationship between $\dot{\mathbf{r}}_j$ and $\dot{\mathbf{q}}$ [Featherstone 2014], we can write:

$$\mathbf{J}_j \dot{\mathbf{q}} = \dot{\mathbf{r}}_j. \quad (8)$$

When the link is in contact with the floor, the velocity perpendicular to the floor has to be zero or positive to prevent penetration. Also, we allow the contact links to have a small tangential velocity σ to prevent an immediate foot motion stop which creates visually unnatural motion. Our contact constraint inequalities read:

$$0 \leq \dot{r}_j^n, \quad |\dot{r}_j^t| \leq \sigma, \quad \text{and} \quad |\dot{r}_j^b| \leq \sigma, \quad (9)$$

where $\dot{r}_j^n$ is the normal component of $\dot{\mathbf{r}}_j$, and $\dot{r}_j^t$ along with $\dot{r}_j^b$ are the tangential elements of $\dot{\mathbf{r}}_j$.

Using the desired acceleration $\ddot{\mathbf{q}}_{des}$ (Eq. (4)), the equation of motion (2), optimal GRF λ estimated in (6) and contact constraints (9), we formulate the optimisation problem for finding the physics-based motion capture result as:

$$\min_{\ddot{\mathbf{q}},\tau} \|\ddot{\mathbf{q}} - \ddot{\mathbf{q}}_{des}\| + \|\tau\|, \\ \text{s.t. } \mathbf{M}\ddot{\mathbf{q}} - \tau = \mathbf{J}^T \mathbf{G}\lambda - \mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}), \text{ and} \quad (10) \\ 0 \leq \dot{r}_j^n, \; |\dot{r}_j^t| \leq \sigma, \; |\dot{r}_j^b| \leq \sigma, \forall j.$$

The first energy term forces the character to reproduce $\mathbf{q}^t_{kin}$. The second energy term is the regulariser that minimises τ to prevent the overshooting, thus modelling natural human-like motion.

After solving (10), the character pose is updated by Eq. (1). We iterate the steps ii) - v) (see stage III in Fig. 3) $n = 4$ times, and stage III returns the n-th output from v) as the final character pose $\mathbf{q}^t_{phys}$. The final output of stage III is a sequence of joint angles and global root translations and rotations that explains the image observations, follows the purely kinematic reconstruction from stage I, yet is physically and anatomically plausible and temporally stable.

5 RESULTS

We first provide implementation details of *PhysCap* (Sec. 5.1) and then demonstrate its qualitative state-of-the-art results (Sec. 5.2). We next evaluate *PhysCap*'s performance quantitatively (Sec. 5.3) and conduct a user study to assess the visual physical plausibility of the results (Sec. 5.4).

We test *PhysCap* on widely-used benchmarks [Habermann et al. 2020; Ionescu et al. 2013; Mehta et al. 2017a] as well as on *backflip* and *jump* sequences provided by [Peng et al. 2018]. We also collect
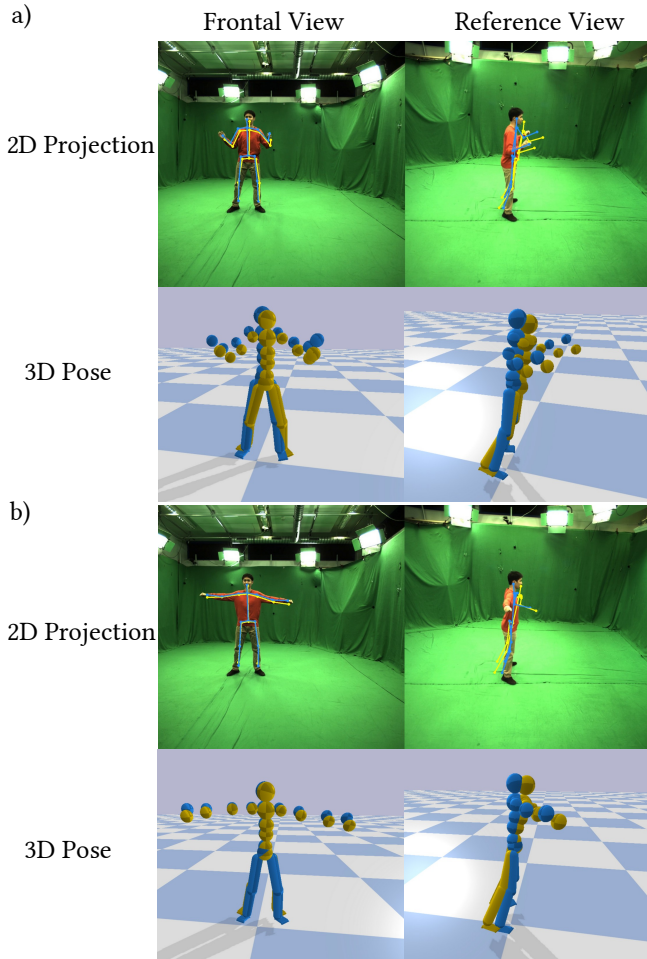
a)

Fig. 6. Two examples of reprojected 3D keypoints obtained by our approach (light blue colour) and Vnect [Mehta et al. 2017b] (yellow colour) together with the corresponding 3D visualisations from different view angles. *PhysCap* produces much more natural and physically plausible postures whereas Vnect suffers from unnatural body leaning (see also the supplementary video).

a new dataset with various challenging motions. It features six sequences in general scenes performed by two subjects[2] recorded at 25 fps. For the recording, we used SONY DSC-RX0, see Table 1 for more details on the sequences.

5.1 Implementation

Our method runs in real time (25 fps on average) on a PC with a Ryzen7 2700 8-Core Processor, 32 GB RAM and GeForce RTX 2070 graphics card. In stage I, we proceed from a freely available demo version of VNect [Mehta et al. 2017b]. Stages II and III are implemented in *python*. In stage II, the network is implemented with *PyTorch* [Paszke et al. 2019]. In stage III, we use the *Rigid Body Dynamics Library* [Felis 2017] to compute dynamic quantities. We

[2]the variety of motions per subject is high; there are only two subjects in the new dataset due to COVID-19 related recording restrictions
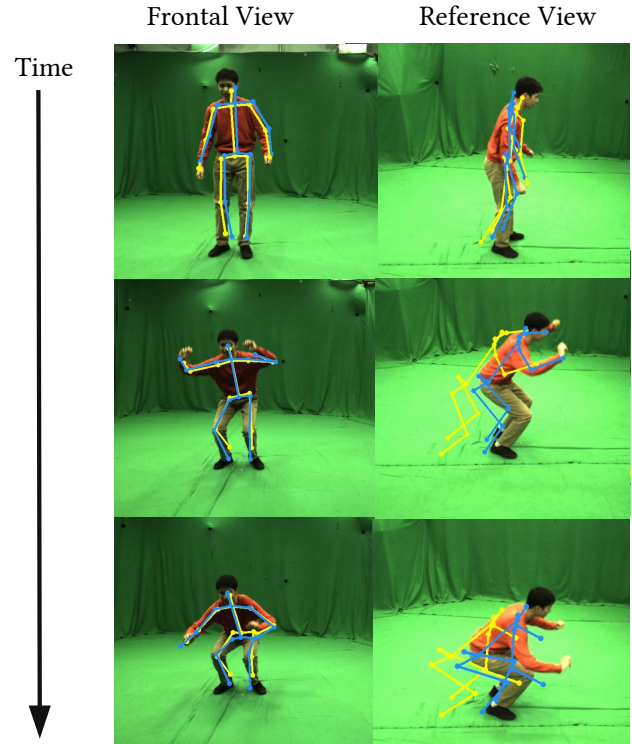

Fig. 7. Reprojected 3D keypoints onto two different images with different view angles for squatting. Frontal view images are used as inputs and images of the reference view are used only for quantitative evaluation. Our results are drawn in light blue, wheres the results by VNect [Mehta et al. 2017b] are provided in yellow. Our reprojections are more feasible, which is especially noticeable in the reference view. See also our supplementary video.

employ the *Pybullet* [Coumans and Bai 2016] as a physics engine for the character motion visualisation and collision detection. In this paper, we set the proportional gain value kp and derivative gain value kd for all joints to 300 and 20, respectively. For the root angular acceleration, kp and kd are set to 340 and 30, respectively. kp and kd of the root linear acceleration are set to 1000 and 80, respectively. These settings are used in all experiments.

5.2 Qualitative Evaluation

The supplementary video and result figures in this paper, in particular Figs. 1 and 11 show that *PhysCap* captures global 3D human poses in real time, even of fast and difficult motions, such as a backflip and a jump, which are of significantly improved quality compared to previous monocular methods. In particular, captured motions are much more temporally stable, and adhere to laws of physics with respect to the naturalness of body postures and fulfilment of environmental constraints, see Figs. 6–8 and 10 for the examples of more natural 3D reconstructions. These properties are essential for many applications in graphics, in particular for stable real-time character animation, which is feasible by directly applying our method's output (see Fig. 1 and the supplementary video).
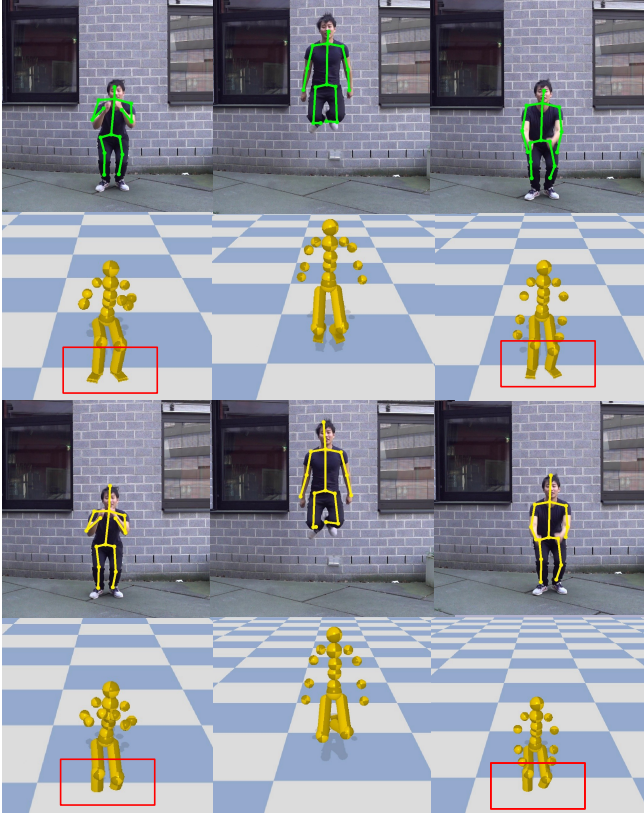
Fig. 8. Several visualisations of the results by our approach and VNect [Mehta et al. 2017b]. The first and second rows show our estimated 3D poses after reprojection in the input image and its 3D view, respectively. Similarly, the third and fourth rows show the reprojected 3D pose and 3D view for VNect. Note that our motion capture shows no foot penetration into the floor plane whereas such artefact is apparent in the VNect results.

5.3 Quantitative Evaluation

In the following, we first describe our evaluation methodology in Sec. 5.3.1. We evaluate *PhysCap* and competing methods under a variety of criteria, *i.e.,* 3D joint position, reprojected 2D joint positions, foot penetration into the floor plane and motion jitter. We compare our approach with current state-of-the-art monocular pose estimation methods,*i.e.,* HMR [Kanazawa et al. 2018], HMMR [Kanazawa et al. 2019] and Vnect [Mehta et al. 2017b] (here we use the so-called *demo version* provided by the authors with further improved accuracy over the original paper due to improved training). For the comparison, we use the benchmark dataset Human3.6M [Ionescu et al. 2013], the DeepCap dataset [Habermann et al. 2020] and MPI-INF-3DHP [Mehta et al. 2017a]. From the Human3.6M dataset, we use the subset of actions that does not have occluding objects in the frame, *i.e., directions, discussions, eating, greeting, posing, purchases, taking photos, waiting, walking, walking dog* and *walking together.* From the DeepCap dataset, we use the subject 2 for this comparison.

5.3.1 Evaluation Methodology. The established evaluation methodology in monocular 3D human pose estimation and capture consists

of testing a method on multiple sequences and reporting the accuracy of 3D joint positions as well as the accuracy of the reprojection into the input views. The accuracy in 3D is evaluated by *mean per joint position error* (MPJPE) in mm, *percentage of correct keypoints* (PCK) and the *area under the receiver operating characteristic (ROC) curve* abbreviated as AUC. The reprojection or mean pixel error e_{2D}^{input} is obtained by projecting the estimated 3D joints onto the input images and taking the average per frame distance to the ground truth 2D joint positions. We report e_{2D}^{input} and its standard deviation denoted by $\sigma_{2D}^{\text{input}}$ with the images of size 1024×1024 pixels.

As explained earlier, these metrics only evaluate limited aspects of captured 3D poses and do not account for essential aspects of temporal stability, smoothness and physical plausibility in reconstructions such as jitter, foot sliding, foot-floor penetration and unnaturally balanced postures. As we show in the supplemental video, top-performing methods on MPJPE and 3D PCK can fare poorly with respect to these criteria. Moreover, MPJPE and PCK are often reported after rescaling of the result in 3D or Procrustes alignment, which further makes these metrics agnostic to the aforementioned artefacts. Thus, we introduce four additional metrics which allow to evaluate the physical plausibility of the results, *i.e., reprojection error to unseen views* e_{2D}^{side}, *motion jitter error* e_{smooth} and two floor penetration errors – *Mean Penetration Error (MPE)* and *Percentage of Non-Penetration (PNP).*

When choosing a reference side view for e_{2D}^{side}, we make sure that the viewing angle between the input and side views has to be sufficiently large, *i.e.,* more than $\sim \frac{\pi}{15}$. Otherwise, if a side view is close to the input view, such effects as unnatural leaning forward can still remain undetected by e_{2D}^{side} in some cases. After reprojection of a 3D structure to an image plane of a side view, all further steps for calculating e_{2D}^{side} are similar to the steps for the standard reprojection error. We also report $\sigma_{2D}^{\text{side}}$, *i.e.,* the standard deviation of e_{2D}^{side}.

To quantitatively compare the motion jitter, we report the deviation of the temporal consistency from the ground truth 3D pose. Our smoothness error e_{smooth} is computed as follows:

$$
\begin{aligned}
Jit_X &= \|\mathbf{p}_X^{s,t} - \mathbf{p}_X^{s,t-1}\|, \\
Jit_{GT} &= \|\mathbf{p}_{GT}^{s,t} - \mathbf{p}_{GT}^{s,t-1}\|, \\
e_{smooth} &= \tfrac{1}{Tm} \textstyle\sum_{t=1}^{T} \sum_{s=1}^{m} |Jit_{GT} - Jit_X|,
\end{aligned}
\tag{11}
$$

where $\mathbf{p}^{s,t}$ represents the 3D position of joint s in the time frame t. T and m denote the total numbers of frames in the video sequence and target 3D joints, respectively. The subscripts X and GT stand for the predicted output and ground truth, respectively. A lower e_{smooth} indicates lower motion jitter in the predicted motion sequence.

MPE and PNP measure the degree of non-physical foot penetration into the ground. MPE is the mean distance between the floor and 3D foot position, and it is computed only when the foot is in contact with the floor. We use the ground truth foot contact labels (Sec. 4.2) to judge the presence of the actual foot contacts. The complementary PNP metric shows the ratio of frames where the feet are not below the floor plane over the entire sequence.

5.3.2 Quantitative Evaluation Results. Table 2 summarises MPJPE, PCK and AUC for root-relative joint positions with (first row) and

Table 2. 3D error comparison on benchmark datasets with VNect [Mehta et al. 2017b], HMR [Kanazawa et al. 2018] and HMMR [Kanazawa et al. 2019]. We report the MPJPE in mm, PCK at 150 mm and AUC. Higher AUC and PCK are better, and lower MPJPE is better. Note that the global root positions for HMR and HMMR were estimated by solving optimisation with a 2D projection loss using the 2D and 3D keypoints obtained from the methods. Our method is on par with and often close to the best-performing approaches on all datasets. It consistently produces the best global root trajectory. As indicated in the text, these widely-used metrics in the pose estimation literature only paint an incomplete picture. For more details, please refer Sec. 5.3.

		DeepCap			Human 3.6M			MPI-INF-3DHP		
		MPJPE [mm] ↓	PCK[%] ↑	AUC[%]↑	MPJPE [mm] ↓	PCK[%] ↑	AUC[%]↑	MPJPE [mm] ↓	PCK[%] ↑	AUC[%]↑
Procrustes	ours	68.9	**95.0**	57.9	65.1	94.8	60.6	104.4	83.9	43.1
	Vnect	**68.4**	94.9	**58.3**	62.7	95.7	61.9	104.5	84.1	43.2
	HMR	77.1	93.8	52.4	**54.3**	96.9	**66.6**	**87.8**	**87.1**	**50.9**
	HMMR	75.5	93.8	53.1	55.0	**96.6**	66.2	106.9	79.5	44.8
no Procrustes	ours	113.0	75.4	39.3	97.4	82.3	46.4	122.9	72.1	35.0
	Vnect	102.4	80.2	**42.4**	89.6	85.1	49.0	**120.2**	**74.0**	**36.1**
	HMR	113.4	75.1	39.0	**78.9**	88.2	**54.1**	130.5	69.7	35.7
	HMMR	**101.4**	**81.0**	42.0	79.4	**88.4**	53.8	174.8	60.4	30.8
global root position	ours	**110.5**	**80.4**	**37.0**	**182.6**	**54.7**	**26.8**	**257.0**	**29.7**	**15.3**
	Vnect	112.6	80.0	36.8	185.1	54.1	26.5	261.0	28.8	15.0
	HMR	251.4	19.5	8.4	204.2	45.8	22.1	505.0	28.6	13.5
	HMMR	213.0	27.7	11.3	231.1	41.6	19.4	926.2	28.0	14.5

Table 3. 2D projection error of a frontal view (input) and side view (non-input) on DeepCap dataset [Habermann et al. 2020]. *PhysCap* performs similarly to VNect on the frontal view, and significantly better on the side view. For further details, see Sec. 5.3 and Fig. 7

	Front View		Side View	
	e_{2D}^{input} [pixel]	$\sigma_{2D}^{\text{input}}$	e_{2D}^{side} [pixel]	$\sigma_{2D}^{\text{side}}$
Ours	21.1	6.7	**35.5**	**16.8**
Vnect [Mehta et al. 2017b]	**14.3**	**2.7**	37.2	18.1

Table 4. Comparison of temporal smoothness on the DeepCap [Habermann et al. 2020] and Human 3.6M datasets [Ionescu et al. 2013]. *PhysCap* significantly outperforms VNect and HMR, and fares comparably to HMMR in terms of this metric. For a detailed explanation, see Sec. 5.3.

		Ours	Vnect	HMR	HMMR
DeepCap	e_{smooth}	**6.3**	11.6	11.7	8.1
	σ_{smooth}	**4.1**	8.6	9.0	5.1
Human 3.6M	e_{smooth}	7.2	11.2	11.2	**6.8**
	σ_{smooth}	6.9	10.1	12.7	**5.9**

Table 5. Comparison of Mean Penetration Error (MPE) and Percentage of Non-Penetration (PNP) on DeepCap dataset [Habermann et al. 2020]. *PhysCap* significantly outperforms VNect on this metric, measuring an essential aspect of physical motion correctness.

	MPE [mm] ↓	σ_{MPE} ↓	PNP [%] ↑
Ours	**28.0**	**25.9**	**92.9**
Vnect [Mehta et al. 2017b]	39.3	37.5	45.6

without (second row) Procrustes alignment before the error computation for our and related methods. We also report the global root position accuracy in the third row. Since HMR and HMMR do not return global root positions as their outputs, we estimate the root translation in 3D by solving an optimisation with 2D projection energy term using the 2D and 3D keypoints obtained from these algorithms (similar to the solution in VNect). The 3D bone lengths of HMR and HMMR were rescaled so that they match the ground truth bone lengths.

In terms of MPJPE, PCK and AUC, our method does not outperform the other approaches consistently but achieves an accuracy that is comparable and often close to the highest on Human3.6M, DeepCap and MPI-INF-3DHP. In the third row, we additionally evaluate the global 3D base position accuracy, which is critical for character animation from the captured data. Here, *PhysCap* consistently outperforms the other methods on all the datasets.

As noted earlier, the above metrics only paint an incomplete picture. Therefore, we also measure the 2D projection errors to the input and side views on the DeepCap dataset, since this dataset includes multiple synchronised views of dynamic scenes with a wide baseline. Table 3 summarises the mean pixel errors e_{2D}^{input} and e_{2D}^{side} together with their standard deviations. In the frontal view, *i.e.*, on e_{2D}^{input}, VNect has higher accuracy than *PhysCap*. However, this comes at the prize of frequently violating physics constraints (floor penetration) and producing unnaturally leaning and jittering 3D poses (see also the supplemental video). In contrast, since *PhysCap*

explicitly models physical pose plausibility, it excels VNect in the side view, which reveals VNect's implausibly leaning postures and root position instability in depth, also see Figs. 6 and 7.

To assess motion smoothness, we report e_{smooth} and its standard deviation σ_{smooth} in Table 4. Our approach outperforms Vnect and HMR by a big margin on both datasets. Our method is better than HMMR on DeepCap dataset and marginally worse on Human3.6M. HMMR is one of the current state-of-the-art algorithms that has an explicit temporal component in the architecture.

Table 5 summarises the MPE and PNP for Vnect and *PhysCap* on DeepCap dataset. Our method shows significantly better results compared to VNect, *i.e.*, about a 30% lower MPE and a by 100% better result in PNP, see Fig. 8 for qualitative examples. Fig. 9 shows plots of contact forces as the functions of time calculated by our approach on the walking sequence from our newly recorded dataset
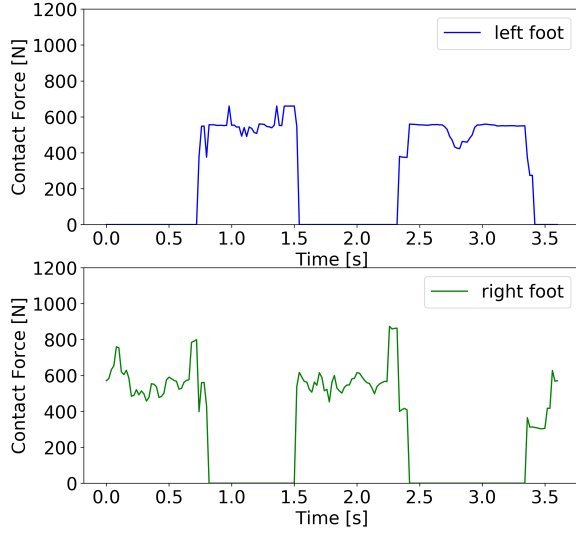
Fig. 9. The estimated contact forces as the functions of time for the walking sequence. We observe that the contact forces remain in a reasonable range for walking motions [Shahabpoor and Pavic 2017].

(sequence 1). The estimated functions fall into a reasonable force range for walking motions [Shahabpoor and Pavic 2017].

5.4 User Study

The notion of *physical plausibility* can be understood and perceived subjectively from person to person. Therefore, in addition to the quantitative evaluation with existing and new metrics, we perform an online user study which allows to subjectively assess and compare the perceived degree of different effects in the reconstructions by a broad audience of people with different backgrounds in computer graphics and vision. In total, we prepared 34 questions with videos, in which we always showed one or two reconstructions at a time (our result, a result by a competing method, or both at the same time). In total, 27 respondents have participated.

There were different types of questions. In 16 questions (category I), the respondents were asked to decide which 3D reconstruction out of two looks more physically plausible to them (the first, the second or undecided). In 12 questions (category II), the respondents were asked to rate how natural the 3D reconstructed motions are or evaluate the degree of an indicated effect (foot sliding, body leaning, *etc.*) on a predefined scale. In five questions (category III), the respondents were also asked to decide which visualisation has a more pronounced indicated artefact. For two questions out of five, 2D projections onto the input 2D image sequence were shown, whereas the remaining questions in this category featured 3D reconstructions. Finally (category IV), the participants were encouraged to list which artefacts in the reconstructions seem to be most apparent and most frequent.

In category I, our reconstructions were preferred in 89.2% of the cases, whereas a competing method was preferred in 1.6% of the cases. Note that at the same time, the decision between the methods has not been made in 8.9% cases. In category II, the respondents

have also found the results of our approach to be significantly more physically plausible than the results of competing methods. The latter were also found to have consistently more jitter, foot sliding and unnatural body leaning. In category III, noteworthy is also that the participants have indicated a higher average perceived accuracy of our reprojections, *i.e.,* 32.7% voted that our results reproject better, whereas the choice felt on the competing methods in 22.6% of the cases. Note that the smoothness and jitter in the results are also reflected in the reprojections, and, thus, both influence how natural the reprojected skeletons look like. At the same time, a high uncertainty of 44.2% indicates that the difference between the reprojections of *PhysCap* and other methods is volatile. For the 3D motions in this category, 82.7% voted that our results show fewer indicated artefacts compared to other approaches, whereas 13.5% of the respondents preferred the competing methods. The decision has not been made in 3.7% of the cases. In category IV, 59% of the participants named jitter as the most frequent and apparent disturbing effect of the competing methods, followed by unnatural body leaning (22%), foot-floor penetration (15%) and foot sliding (15%).

The user study confirms a high level of physical plausibility and naturalness of *PhysCap* results. We see that also subjectively, a broad audience coherently finds our results of high visual quality, and the gap to the competing methods is substantial. This strengthens our belief about the suitability of *PhysCap* for computer graphics and primarily virtual character animation in real time.

6 DISCUSSION

Our physics-based monocular 3D human motion capture algorithm significantly reduces the common artefacts of other monocular 3D pose estimation methods such as motion jitter, penetration into the floor, foot sliding and unnatural body leaning. The experiments have shown that our state prediction network generalises well across scenes with different backgrounds (see Fig. 11). However, in the case of foot occlusion, our state prediction network can sometimes mispredict the foot contact states, resulting in the erroneous hard zero velocity constraint for feet. Additionally, our approach requires the calibrated floor plane to apply the foot contact constraint effectively; standard calibration techniques can be used for this.

Swift motions can be challenging for stage I of our pipeline, which can cause inaccuracies in the estimates of the subsequent stages, as well as in the final estimate. In future, other monocular kinematic pose estimators than [Mehta et al. 2017b] could be tested in stage I, in case they are trained to handle occlusions and very fast motions better. Moreover, note that – although we use a single parameter set for *PhysCap* in all our experiments (see Sec. 5) – users can adjust the quality of the reconstructed motions by tuning the gain parameters of PD controller depending on the scenario. By increasing the derivative gain value, the reconstructed poses are smoother, which, however, can cause motion delay compared to the input video, especially when the observed motions are very fast. By reducing the derivative gain value, our optimisation with a virtual character can track image sequence with less motion delay, at the cost of less temporally coherent motion. We demonstrate this trade-off in the supplemental video.
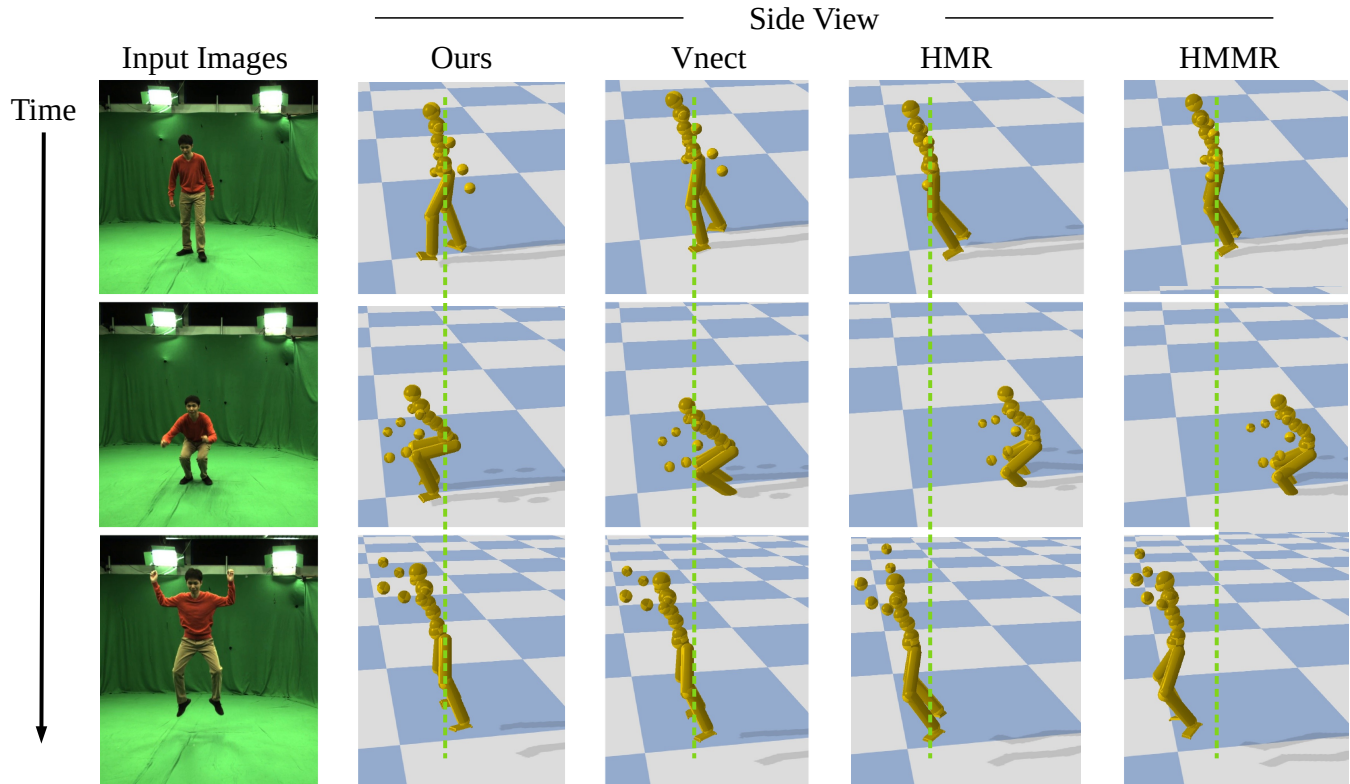
Fig. 10. Several side (non-input) view visualisations of the results by our approach, Vnect [Mehta et al. 2017b], HMR [Kanazawa et al. 2018] and HMMR [Kanazawa et al. 2019] on DeepCap dataset. The green dashed lines indicate the expected root positions over time. It is apparent from the side view that our *PhysCap* does not suffer from the unnatural body sliding along the depth direction, unlike other approaches. The global base positions for HMR and HMMR were computed by us using the root-relative predictions of these techniques, see Sec. 5.3.2 for more details.

Further, while our method works in front of general backgrounds, we assume there is a ground plane in the scene, which is the case for most man-made environments, but not irregular outdoor terrains. Finally, our method currently only considers a subset of potential body-to-environment contacts in a physics-based way. As part of future work, we will investigate explicit modelling of self-collisions, as well as hand-scene interactions or contacts of legs and body in sitting and lying poses.

7 CONCLUSIONS

We have presented *PhysCap* – the first physics-based approach for a global 3D human motion capture from a single RGB camera that runs in real time at 25 fps. Thanks to the pose optimisation framework using PD joint control, the results of *PhysCap* evince improved physical plausibility, temporal consistency and significantly fewer artefacts such as jitter, foot sliding, unnatural body leaning and foot-floor penetration, compared to other existing approaches (some of them include temporal constraints). We also introduced new error metrics to evaluate these improved properties which are not easily captured by metrics used in the established pose estimation benchmarks. Moreover, our user study further confirmed these

improvements. In future work, our algorithm can be extended for various contact positions (not only the feet).

REFERENCES

Farhan A. Salem and Ayman Aly. 2015. PD Controller Structures: Comparison and Selection for an Electromechanical System. *International Journal of Intelligent Systems and Applications (IJISA)* 7, 2 (2015), 1–12.

Adobe. 2020. Mixamo. https://www.mixamo.com/. Accessed: 2020-04-15.

Sheldon Andrews, Ivan Huerta, Taku Komura, Leonid Sigal, and Kenny Mitchell. 2016. Real-Time Physics-Based Motion Capture with Sparse Sensors. In *European Conference on Visual Media Production (CVMP)*.

Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2014. Human Pose Estimation: New Benchmark and State of the Art Analysis. In *Computer Vision and Pattern Recognition (CVPR)*.

Ronen Barzel, John F. Hughes, and Daniel N. Wood. 1996. Plausible Motion Simulation for Computer Graphics Animation. In *Proceedings of the Eurographics Workshop on Computer Animation and Simulation*.

Kevin Bergamin, Simon Clavet, Daniel Holden, and James Richard Forbes. 2019. DReCon: Data-Driven Responsive Control of Physics-Based Characters. *ACM Transactions On Graphics (TOG)* 38, 6 (2019).

Liefeng Bo and Cristian Sminchisescu. 2010. Twin Gaussian Processes for Structured Prediction. *International Journal of Computer Vision (IJCV)* 87 (2010), 28–52.

Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J. Black. 2016. Keep it SMPL: Automatic Estimation of 3D Human Pose and Shape from a Single Image. In *European Conference on Computer Vision (ECCV)*.

Derek Bradley, Tiberiu Popa, Alla Sheffer, Wolfgang Heidrich, and Tamy Boubekeur. 2008. Markerless garment capture. *ACM Transactions on Graphics (TOG)* 27, 3 (2008), 99.

Ernesto Brau and Hao Jiang. 2016. 3D Human Pose Estimation via Deep Learning from 2D Annotations. In *International Conference on 3D Vision (3DV)*.
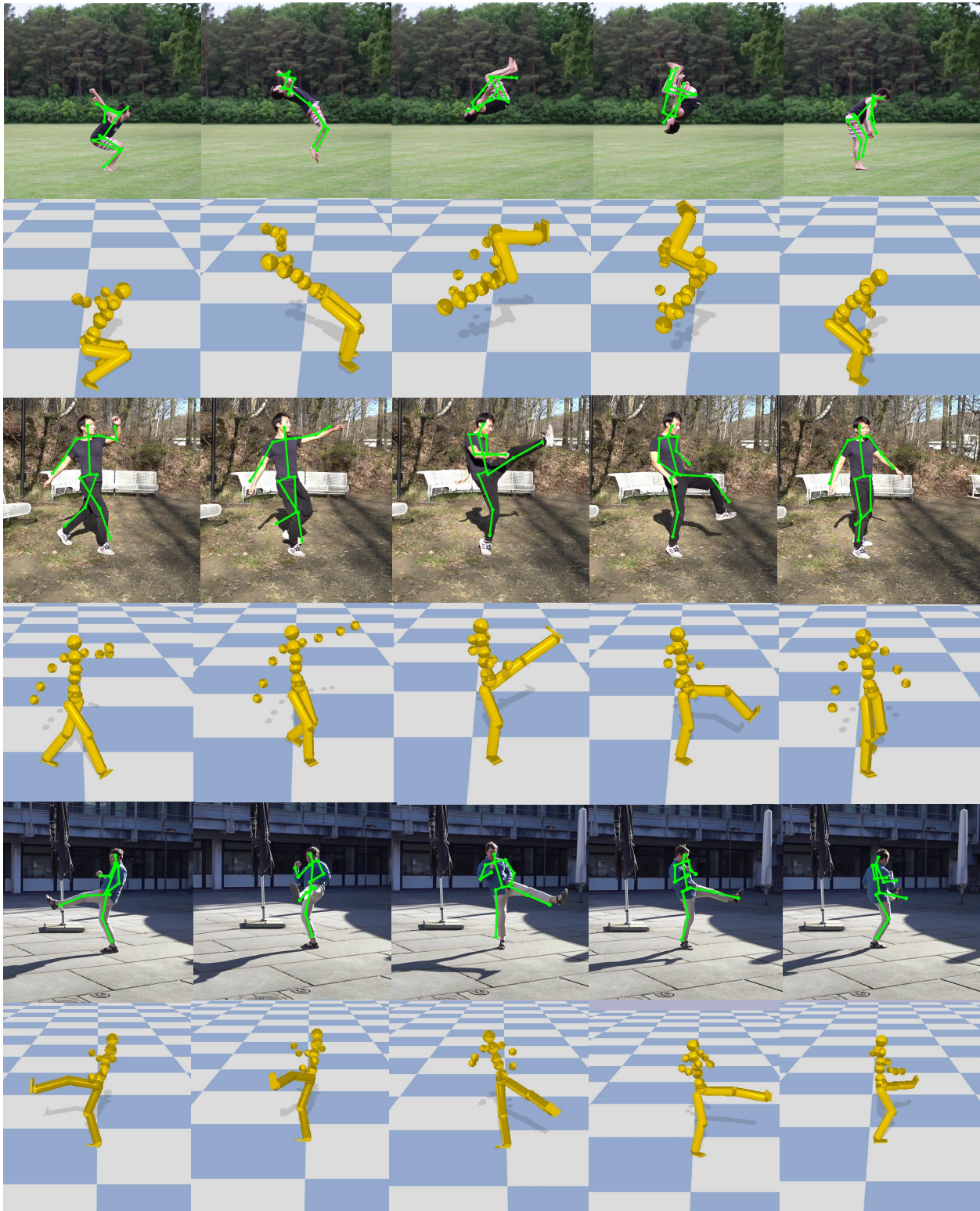
Fig. 11. Representative 2D reprojections and the corresponding 3D poses of our *PhysCap* approach. Note that, even with the challenging motions, our global poses in 3D have high quality and 2D reprojections to the input images are accurate as well. See our supplementary video for more results on these sequences. The *backflip* video in the first row is taken from [Peng et al. 2018]. Other sequences are from our own recordings.

Thomas Brox, Bodo Rosenhahn, Juergen Gall, and Daniel Cremers. 2010. Combined region- and motion-based 3D tracking of rigid and articulated objects. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 32, 3 (2010), 402–415.

Cedric Cagniart, Edmond Boyer, and Slobodan Ilic. 2010. Free-Form Mesh Tracking: a Patch-Based Approach. In *Computer Vision and Pattern Recognition (CVPR)*.

Géry Casiez, Nicolas Roussel, and Daniel Vogel. 2012. 1 € Filter: A Simple Speed-Based Low-Pass Filter for Noisy Input in Interactive Systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*.

Ching-Hang Chen and Deva Ramanan. 2017. 3D Human Pose Estimation = 2D Pose Estimation + Matching. In *Computer Vision and Pattern Recognition (CVPR)*.

Stelian Coros, Philippe Beaudoin, and Michiel van de Panne. 2010. Generalized Biped Walking Control. *ACM Transactions On Graphics (TOG)* 29, 4 (2010).

Erwin Coumans and Yunfei Bai. 2016. Pybullet, a python module for physics simulation for games, robotics and machine learning. *GitHub repository* (2016).

Rishabh Dabral, Anurag Mundhada, Uday Kusupati, Safeer Afaque, Abhishek Sharma, and Arjun Jain. 2018. Learning 3D Human Pose from Structure and Motion. In *European Conference on Computer Vision (ECCV)*.

Edilson De Aguiar, Carsten Stoll, Christian Theobalt, Naveed Ahmed, Hans-Peter Seidel, and Sebastian Thrun. 2008. Performance Capture from Sparse Multi-View Video. *ACM Transactions on Graphics (TOG)* 27, 3 (2008).

Ahmed Elhayek, Edilson de Aguiar, Arjun Jain, Jonathan Thompson, Leonid Pishchulin, Mykhaylo Andriluka, Christoph Bregler, Bernt Schiele, and Christian Theobalt. 2016. MARCOnI—ConvNet-Based MARker-Less Motion Capture in Outdoor and Indoor Scenes. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 39, 3 (2016), 501–514.

Ahmed Elhayek, Carsten Stoll, Kwang In Kim, and Christian Theobalt. 2014. Outdoor Human Motion Capture by Simultaneous Optimization of Pose and Camera Parameters. *Computer Graphics Forum* (2014).

Petros Faloutsos, Michiel van de Panne, and Demetri Terzopoulos. 2001. Composable Controllers for Physics-Based Character Animation. In *Annual Conference on Computer Graphics and Interactive Techniques*. 251–260.

Roy Featherstone. 2014. *Rigid body dynamics algorithms*.

Martin L. Felis. 2017. RBDL: an Efficient Rigid-Body Dynamics Library using Recursive Algorithms. *Autonomous Robots* 41, 2 (2017), 495–511.

Juergen Gall, Bodo Rosenhahn, Thomas Brox, and Hans-Peter Seidel. 2010. Optimization and Filtering for Human Motion Capture - a Multi-Layer Framework. *International Journal of Computer Vision (IJCV)* 87, 1 (2010), 75–92.

Juergen Gall, Carsten Stoll, Edilson De Aguiar, Christian Theobalt, Bodo Rosenhahn, and Hans-Peter Seidel. 2009. Motion Capture Using Joint Skeleton Tracking and Surface Estimation. In *Computer Vision and Pattern Recognition (CVPR)*.

Marc Habermann, Weipeng Xu, Michael Zollhofer, Gerard Pons-Moll, and Christian Theobalt. 2020. DeepCap: Monocular Human Performance Capture Using Weak Supervision. In *Computer Vision and Pattern Recognition (CVPR)*.

Marc Habermann, Weipeng Xu, Michael Zollhöfer, Gerard Pons-Moll, and Christian Theobalt. 2019. LiveCap: Real-Time Human Performance Capture From Monocular Video. *ACM Transactions On Graphics (TOG)* 38, 2 (2019), 14:1–14:17.

Ikhsanul Habibie, Weipeng Xu, Dushyant Mehta, Gerard Pons-Moll, and Christian Theobalt. 2019. In the Wild Human Pose Estimation Using Explicit 2D Features and Intermediate 3D Representations. In *Computer Vision and Pattern Recognition (CVPR)*.

Mohamed Hassan, Vasileios Choutas, Dimitrios Tzionas, and Michael J. Black. 2019. Resolving 3D Human Pose Ambiguities with 3D Scene Constraints. In *International Conference on Computer Vision (ICCV)*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *Computer Vision and Pattern Recognition (CVPR)*.

Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. 2013. Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 36, 7 (2013), 1325–1339.

Yifeng Jiang, Tom Van Wouwe, Friedl De Groote, and C. Karen Liu. 2019. Synthesis of Biologically Realistic Human Motion Using Joint Torque Actuation. *ACM Transactions On Graphics (TOG)* 38, 4 (2019).

S. Johnson and M. Everingham. 2011. Learning Effective Human Pose Estimation from Inaccurate Annotation. In *Computer Vision and Pattern Recognition (CVPR)*.

Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. 2018. End-to-end Recovery of Human Shape and Pose. In *Computer Vision and Pattern Recognition (CVPR)*.

Angjoo Kanazawa, Jason Y. Zhang, Panna Felsen, and Jitendra Malik. 2019. Learning 3D Human Dynamics from Video. In *Computer Vision and Pattern Recognition (CVPR)*.

Muhammed Kocabas, Nikos Athanasiou, and Michael J. Black. 2020. VIBE: Video Inference for Human Body Pose and Shape Estimation. In *Computer Vision and Pattern Recognition (CVPR)*.

Onorina Kovalenko, Vladislav Golyanik, Jameel Malik, Ahmed Elhayek, and Didier Stricker. 2019. Structure from Articulated Motion: Accurate and Stable Monocular 3D Reconstruction without Training Data. *Sensors* 19, 20 (2019).

Seunghwan Lee, Moonseok Park, Kyoungmin Lee, and Jehee Lee. 2019. Scalable Muscle-Actuated Human Simulation and Control. *ACM Transactions On Graphics (TOG)* 38, 4 (2019).

Kenneth Levenberg. 1944. A method for the solution of certain non-linear problems in least squares. *Quarterly Journal of Applied Mathmatics* II, 2 (1944), 164–168.

Sergey Levine and Jovan Popović. 2012. Physically Plausible Simulation for Character Animation. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*.

Zongmian Li, Jiri Sedlar, Justin Carpentier, Ivan Laptev, Nicolas Mansard, and Josef Sivic. 2019. Estimating 3D Motion and Forces of Person-Object Interactions from Monocular Video. In *Computer Vision and Pattern Recognition (CVPR)*.

Libin Liu, KangKang Yin, Michiel van de Panne, Tianjia Shao, and Weiwei Xu. 2010. Sampling-Based Contact-Rich Motion Control. *ACM Transactions On Graphics (TOG)* 29, 4 (2010), 128:1–128:10.

Yebin Liu, Carsten Stoll, Juergen Gall, Hans-Peter Seidel, and Christian Theobalt. 2011. Markerless Motion Capture of Interacting Characters using Multi-View Image Segmentation. In *Computer Vision and Pattern Recognition (CVPR)*.

Adriano Macchietto, Victor Zordan, and Christian R. Shelton. 2009. Momentum Control for Balance. In *ACM SIGGRAPH*.

Donald W. Marquardt. 1963. An Algorithm for Least-Squares Estimation of Nonlinear Parameters. *SIAM J. Appl. Math.* 11, 2 (1963), 431–441.

Ricardo Martin-Brualla, Rohit Pandey, Shuoran Yang, Pavel Pidlypenskyi, Jonathan Taylor, Julien Valentin, Sameh Khamis, Philip Davidson, Anastasia Tkach, Peter Lincoln, Adarsh Kowdle, Christoph Rhemann, Dan B Goldman, Cem Keskin, Steve Seitz, Shahram Izadi, and Sean Fanello. 2018. LookinGood: Enhancing Performance Capture with Real-Time Neural Re-Rendering. *ACM Transactions On Graphics (TOG)* 37, 6 (2018).

Julieta Martinez, Rayat Hossain, Javier Romero, and James J. Little. 2017. A Simple Yet Effective Baseline for 3D Human Pose Estimation. In *International Conference on Computer Vision (ICCV)*.

Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. 2017a. Monocular 3D Human Pose Estimation In The Wild Using Improved CNN Supervision. In *International Conference on 3D Vision (3DV)*.

Dushyant Mehta, Oleksandr Sotnychenko, Franziska Mueller, Weipeng Xu, Mohammad Elgharib, Hans-Peter Seidel, Helge Rhodin, Gerard Pons-Moll, and Christian Theobalt. 2020. XNect: Real-time Multi-Person 3D Motion Capture with a Single RGB Camera. *ACM Transactions on Graphics (TOG)* 39, 4.

Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. 2017b. VNect: Real-time 3D Human Pose Estimation with a Single RGB Camera. *ACM Transactions on Graphics* 36, 4, 14.

Aron Monszpart, Paul Guerrero, Duygu Ceylan, Ersin Yumer, and Niloy J. Mitra. 2019. IMapper: Interaction-Guided Scene Mapping from Monocular Videos. *ACM Transactions On Graphics (TOG)* 38, 4 (2019).

Francesc Moreno-Noguer. 2017. 3D Human Pose Estimation From a Single Image via Distance Matrix Regression. In *Computer Vision and Pattern Recognition (CVPR)*.

Shin'ichiro Nakaoka, Atsushi Nakazawa, Fumio Kanehiro, Kenji Kaneko, Mitsuharu Morisawa, Hirohisa Hirukawa, and Katsushi Ikeuchi. 2007. Learning from Observation Paradigm: Leg Task Models for Enabling a Biped Humanoid Robot to Imitate Human Dances. *The International Journal of Robotics Research* 26, 8 (2007), 829–844.

Alejandro Newell, Kaiyu Yang, and Jia Deng. 2016. Stacked Hourglass Networks for Human Pose Estimation. In *European Conference on Computer Vision (ECCV)*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems (NeurIPS)*. 8026–8037.

Georgios Pavlakos, Xiaowei Zhou, and Kostas Daniilidis. 2018. Ordinal Depth Supervision for 3D Human Pose Estimation. In *Computer Vision and Pattern Recognition (CVPR)*.

Georgios Pavlakos, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis. 2017. Coarse-to-Fine Volumetric Prediction for Single-Image 3D Human Pose. In *Computer Vision and Pattern Recognition (CVPR)*.

Xue Bin Peng, Angjoo Kanazawa, Jitendra Malik, Pieter Abbeel, and Sergey Levine. 2018. SFV: Reinforcement Learning of Physical Skills from Videos. *ACM Transactions On Graphics (TOG)* 37, 6 (2018).

Helge Rhodin, Mathieu Salzmann, and Pascal Fua. 2018. Unsupervised Geometry-Aware Representation Learning for 3D Human Pose Estimation. In *European Conference on Computer Vision (ECCV)*.

Grégory Rogez, Philippe Weinzaepfel, and Cordelia Schmid. 2019. LCR-Net++: Multi-Person 2D and 3D Pose Detection in Natural Images. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2019).

Erfan Shahabpoor and Aleksandar Pavic. 2017. Measurement of Walking Ground Reactions in Real-Life Environments: A Systematic Review of Techniques and Technologies. *Sensors* 17, 9 (2017), 2085.

Dana Sharon and Michiel van de Panne. 2005. Synthesis of Controllers for Stylized Planar Bipedal Walking. In *International Conference on Robotics and Animation (ICRA)*.

Jonathan Starck and Adrian Hilton. 2007. Surface capture for performance-based animation. *IEEE Computer Graphics and Applications (CGA)* 27, 3 (2007), 21–31.

Carsten Stoll, Nils Hasler, Juergen Gall, Hans-Peter Seidel, and Christian Theobalt. 2011. Fast articulated motion tracking using a sums of Gaussians body model. In *International Conference on Computer Vision (ICCV)*.

Bugra Tekin, Isinsu Katircioglu, Mathieu Salzmann, Vincent Lepetit, and Pascal Fua. 2016. Structured Prediction of 3D Human Pose with Deep Neural Networks. In *British Machine Vision Conference (BMVC)*.

Denis Tomè, Chris Russell, and Lourdes Agapito. 2017. Lifting from the Deep: Convolutional 3D Pose Estimation from a Single Image. In *Computer Vision and Pattern Recognition (CVPR)*.

Daniel Vlasic, Ilya Baran, Wojciech Matusik, and Jovan Popović. 2008. Articulated mesh animation from multi-view silhouettes. In *ACM Transactions on Graphics (TOG)*, Vol. 27. 97.

Daniel Vlasic, Pieter Peers, Ilya Baran, Paul Debevec, Jovan Popović, Szymon Rusinkiewicz, and Wojciech Matusik. 2009. Dynamic Shape Capture using Multi-View Photometric Stereo. *ACM Transactions on Graphics (TOG)* 28, 5 (2009), 174.

Marek Vondrak, Leonid Sigal, Jessica Hodgins, and Odest Jenkins. 2012. Video-based 3D Motion Capture Through Biped Control. *ACM Transactions On Graphics (TOG)* 31, 4 (2012), 1–12.

Bastian Wandt and Bodo Rosenhahn. 2019. RepNet: Weakly Supervised Training of an Adversarial Reprojection Network for 3D Human Pose Estimation. In *Computer Vision and Pattern Recognition (CVPR)*.

Yangang Wang, Yebin Liu, Xin Tong, Qionghai Dai, and Ping Tan. 2018. Robust Non-rigid Motion Tracking and Surface Reconstruction Using L0 Regularization. *IEEE Transctions on Visualization and Computer Graphics (TVCG)* 24, 5 (2018), 1770–1783.

Michael Waschbüsch, Stephan Würmlin, Daniel Cotting, Filip Sadlo, and Markus Gross. 2005. Scalable 3D Video of Dynamic Scenes. *The Visual Computer* 21, 8-10 (2005), 629–638.

Xiaolin Wei and Jinxiang Chai. 2010. Videomocap: Modeling Physically Realistic Human Motion from Monocular Video Sequences. In *ACM Transactions on Graphics (TOG)*, Vol. 29.

Pawel Wrotek, Odest Chadwicke Jenkins, and Morgan McGuire. 2006. Dynamo: Dynamic, Data-Driven Character Control with Adjustable Balance. In *ACM Sandbox Symposium on Video Games 2006*.

Chenglei Wu, Kiran Varanasi, and Christian Theobalt. 2012. Full Body Performance Capture under Uncontrolled and Varying Illumination: A Shading-Based Approach. In *European Conference on Computer Vision (ECCV)*.

Lan Xu, Weipeng Xu, Vladislav Golyanik, Marc Habermann, Lu Fang, and Christian Theobalt. 2020. EventCap: Monocular 3D Capture of High-Speed Human Motions using an Event Camera. In *Computer Vision and Pattern Recognition (CVPR)*.

Wei Yang, Wanli Ouyang, Xiaolong Wang, Jimmy Ren, Hongsheng Li, and Xiaogang Wang. 2018. 3D Human Pose Estimation in the Wild by Adversarial Learning. In *Computer Vision and Pattern Recognition (CVPR)*.

Andrei Zanfir, Elisabeta Marinoiu, and Cristian Sminchisescu. 2018. Monocular 3D Pose and Shape Estimation of Multiple People in Natural Scenes - The Importance of Multiple Scene Constraints. In *Computer Vision and Pattern Recognition (CVPR)*.

Petrissa Zell, Bastian Wandt, and Bodo Rosenhahn. 2017. Joint 3D Human Motion Capture and Physical Analysis from Monocular Videos. In *Computer Vision and Pattern Recognition (CVPR) Workshops*.

Peizhao Zhang, Kristin Siu, Jianjie Zhang, C Karen Liu, and Jinxiang Chai. 2014. Leveraging Depth Cameras and Wearable Pressure Sensors for Full-Body Kinematics and Dynamics Capture. *ACM Transactions on Graphics (TOG)* 33, 6 (2014), 1–14.

Yuxiang Zhang, Liang An, Tao Yu, xiu Li, Kun Li, and Yebin Liu. 2020. 4D Association Graph for Realtime Multi-Person Motion Capture Using Multiple Video Cameras. In *International Conference on Computer Vision (ICCV)*.

Yu Zheng and Katsu Yamane. 2013. Human Motion Tracking Control with Strict Contact Force Constraints for Floating-Base Humanoid Robots. In *International Conference on Humanoid Robots (Humanoids)*.

Xingyi Zhou, Qixing Huang, Xiao Sun, Xiangyang Xue, and Yichen Wei. 2017. Towards 3D Human Pose Estimation in the Wild: A Weakly-Supervised Approach. In *International Conference on Computer Vision (ICCV)*.

Xingyi Zhou, Xiao Sun, Wei Zhang, Shuang Liang, and Yichen Wei. 2016. Deep Kinematic Pose Regression. In *European Conference on Computer Vision (ECCV)*.

Yuliang Zou, Jimei Yang, Duygu Ceylan, Jianming Zhang, Federico Perazzi, and Jia-Bin Huang. 2020. Reducing Footskate in Human Motion Reconstruction with Ground Contact Constraints. In *Winter Conference on Applications of Computer Vision (WACV)*.