Fully Automatic Multi-Object Articulated Motion Tracking

Ahmed Elhayek*

Artificial Intelligence department, University of Prince Mugrin (UPM), Madinah, Saudi Arabia

* E-mail of the corresponding author: a.elhayek@upm.sa

Abstract

Fully automatic tracking of articulated motion in real-time with a monocular RGB camera is a challenging problem which is essential for many virtual reality (VR) and human-computer interaction applications. In this paper, we present an algorithm for multiple articulated objects tracking based on monocular RGB image sequence. Our algorithm can be directly employed in practical applications as it is fully automatic, real-time, and temporally stable. It consists of the following stages: dynamic objects counting, objects specific 3D skeletons generation, initial 3D poses estimation, and 3D skeleton fitting which fits each 3D skeleton to the corresponding 2D body-parts locations. In the skeleton fitting stage, the 3D pose of every object is estimated by maximizing an objective function that combines a skeleton fitting term with motion and pose priors. To illustrate the importance of our algorithm for practical applications, we present competitive results for real-time tracking of multiple humans. Our algorithm detects objects that enter or leave the scene, and dynamically generates or deletes their 3D skeletons. This makes our monocular RGB method optimal for real-time applications. We show that our algorithm is applicable for tracking multiple objects in outdoor scenes, community videos, and low-quality videos captured with mobile-phone cameras.

Keywords: Multi-object motion tracking, Articulated motion capture, Deep learning, Anthropometric data, 3D pose estimation.

DOI: 10.7176/CEIS/12-1-01
Publication date: March 31st 2021

1. Introduction

Multiple articulated objects tracking algorithms have applications in many fields such as Virtual reality (VR), augmented reality (AR), 3D character animation (i.e. for movies and games), human-computer interaction, and sports. Tracking articulated objects from a single RGB image sequence is a challenging computer vision problem. Practical applications require real-time inference with low latency which makes this problem even harder. This problem is challenging for many other reasons such as self-similar parts, each articulated object has many degrees of freedom (DOF), occlusion by other objects, and self-occlusion.

Recently, significant progress has been achieved in articulated object tracking approaches. The new algorithms have the ability to estimate both pose and shape (e.g. mesh) from RGB images [1–4]. Although many marker-less algorithms have achieved high accuracy under challenging conditions, most commercial VR systems still use marker-based methods that require placing markers on the human body. One of the main reasons is that marker-less algorithms require several manual initialization steps (e.g. estimation of objects number, their 3D models, and their initial poses) which are cumbersome, require a lot of experience, and time-consuming. Besides, most marker-less approaches [5–8] fail to reliably track articulated motion in general scenes with a single RGB camera. Therefore, our goal is to design a monocular fully automatic algorithm. Many recent algorithms have managed to estimate accurate human motion from monocular depth (i.e. RGB-D) cameras [9–11]. However, the RGB-D cameras fail in general outdoor scenes due to sunlight interference. Moreover, these cameras have lower resolution, limited range, and higher power consumption.

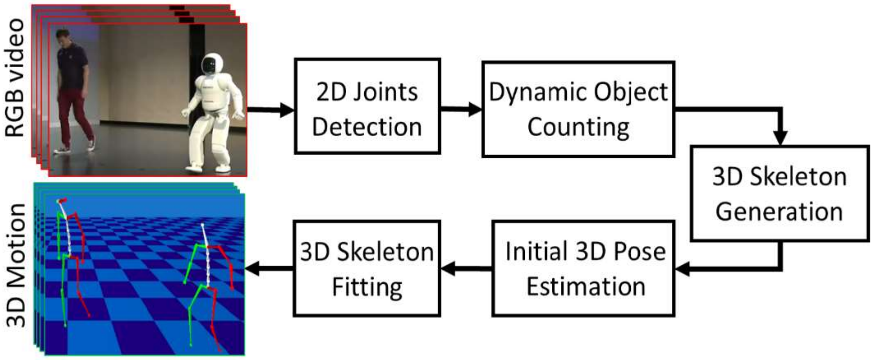


Fig 1. Fully automatic multiple articulated objects tracking pipeline overview: Given 2D joints position, our method dynamically detects the change in the number of objects. Then, it generates a specific 3D skeleton for each object and estimates its initial 3D pose. The final stage fits each 3D skeleton to the corresponding 2D body-parts locations.

Monocular RGB cameras are very common in laptops and smartphones. Thus, developing a fully automatic real-time multi-object marker-less human motion tracking algorithm that works with monocular cameras is essential for many applications. An example of these applications is to track multiple objects (e.g. persons) and to include their animation in VR environment using the camera of a VR-headset. Furthermore, this algorithm allows to interface PCs, laptops, or smartphones using their cameras (e.g. play games). Developing such an algorithm requires 1) automatic estimation of the number of objects in the scene 2) automatic generation of their 3D skeletons 3) automatic estimation of their initial 3D location 4) dynamic generation or deletion of the 3D skeletons for objects entering or leaving the scene 5) real-time multi-object 3D pose estimation.

Recently, the deep learning-based 2D joints positions (i.e. 2D pose) estimation algorithms achieved very high accuracy. In this paper, we utilize the progress in 2D pose estimation to develop a fully automatic multi-object 3D motion tracking algorithm that meets all these requirements. An overview of the proposed pipeline is shown in Fig. 1.

Although some of the state-of-the-art algorithms achieve better accuracy than our algorithm, they fail under our challenging multi-object tracking conditions. Many of these algorithms [1, 3] track only a single object (e.g. [3]). Moreover, other monocular algorithms such as [12, 13] are offline and exhibit jitter over time due to per-frame estimation. Our algorithm performs automatic object-specific skeleton generation and initial pose localization of a varying number of objects in real-time. It can track multiple objects moving in front of cluttered and non-static backgrounds with a handheld camera which suffers from high distortion. It also succeeds in case of strong illumination changes. It works with any mobile-phone cameras or webcams. It can also track human motion in community videos (e.g. YouTube videos); see Fig. 4. The estimated multi-object motions can be used in many fields such as Augmented reality, Virtual Reality, motion-driven 3D game character control, and human-computer interaction. Furthermore, our algorithm can be optimized for smartphones and driving assistance applications. In our experiments, we show that our algorithm can track even complex and fast body motion of multi-object in real-time. We managed to track complex motions of multiple articulated objects in outdoor scenes with a moving mobile phone camera, and a webcam in an office.

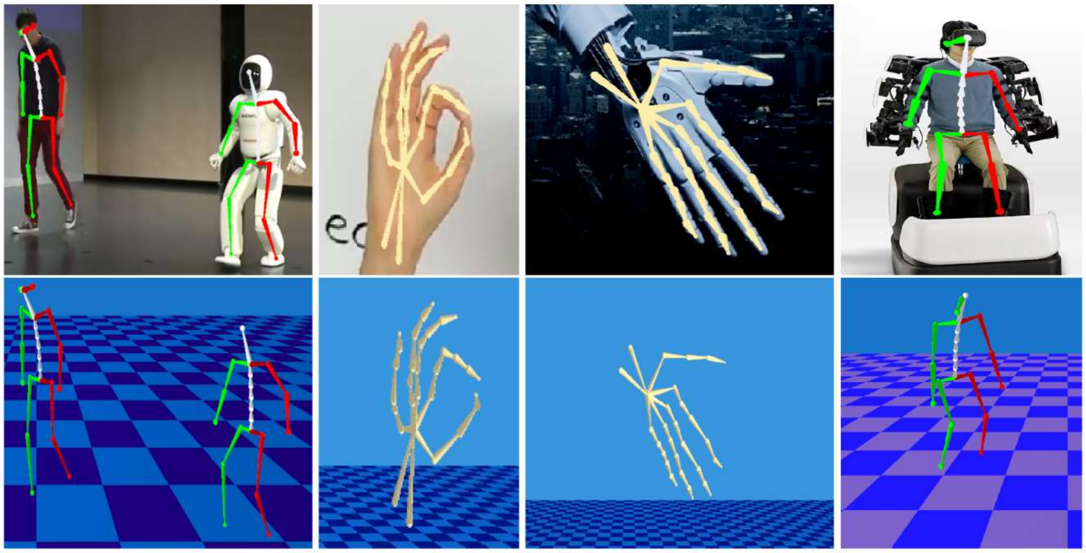


Fig 2. Our algorithm recovers 3D skeletons poses in real-time. It tracks complex motions of a human and a robot in a community video (left), estimates poses of hands (middle) and tracks a person in VR environment (right). Top row shows overlaid 2D skeletons and bottom row shows 3D visualizations of the tracked objects.

Our algorithmic contributions that enabled achieving such results, are:

1. Real-time, simple, and automatic multi-object 3D skeletons generation; see Section 4.1.
2. Automatic initial 3D location estimation of each object in the scene; see Section 4.2.
3. Automatic detection of the change in the number of objects and generating or deleting the corresponding 3D skeletons on the fly while tracking; see Section 4.3.
4. An algorithm which tracks articulated joint angles of multiple objects at high accuracy and temporal stability in real-time, given 2D body-part locations; see Section 4.3.

In the rest of the paper, we focus on tracking human motion. Nevertheless, since the human represents a difficult type of articulated object, our algorithm is applicable to a wide range of articulated subjects. Fig. 2 shows the result of applying our pipeline to other articulated object given their corresponding 2D joints detections. This illustrates the generalizability of our algorithm which can track any articulated object given 1) 2D joint positions estimation algorithm for this object (e.g. CNN-based pose estimation algorithm) and 2) existing relation between the lengths of the object parts (e.g. anthropometric data).

Moreover, our method still generates plausible tracking results even when the relation between the object parts is fixed to rough estimate; see section 5. The core of this approach was presented in [14], but is further generalized in this article to work for tracking any kind of articulated objects. An overview of the generalized pipeline is shown in Fig. 1.

2. RELATED WORK

It is difficult to cover the related works of all possible articulated objects. Thus, we refer the reader to [15, 16] for a comprehensive review of articulated objects such as the human hand. As the human is the most common articulated object and to avoid making this section too long, we summarized the related human articulated motion tracking algorithms.

Human motion tracking has seen great progress in recent years. We refer the reader to the surveys [17–19] for an overview. Most generative multi-view human motion tracking methods rely on maximizing the similarity between the input images and the 3D human model projection. These methods are slow but achieve reasonable accuracy because they exploit the temporal smoothness of the motion [20–26]. Monocular Human motion tracking is more applicable in many fields. Thus, we focus the discussion in the rest of this section on methods that rely on a monocular RGB camera.

Depth-based motion tracking methods [10, 11] have achieved robust real-time results. However, in this section, we focus on RGB-based methods. These methods can be divided into generative and discriminative methods. The generative motion tracking problem is fundamentally under-constrained in the case of monocular input. Thus, it is only successful for motion tracking from short clips and when combined with strong motion priors [27]. Manual annotation and correction of frames is suitable for some applications such as actor reshaping in movies [28] and garment replacement in videos [29]. These generative algorithms preclude live applications because of manual interaction and expensive optimization.

Recently, many monocular discriminative human pose estimation algorithms have been introduced. Some of them discriminatively learned the mapping from the image directly to human joint locations [1, 2, 30–34]. CNN-based 2D and 3D human pose estimation approaches achieved high accuracy. For instance, [35–38] estimate human 3D pose directly from monocular image or video. Chen et al. [39] synthesize training images automatically with ground truth pose annotations and train CNNs with these synthetic images for 3D pose estimation.

Other approaches estimate 3D human pose from 2D body parts locations in a monocular image [40, 41] or image sequence [42–47]. Many of these works rely on manually labelled 2D body part locations. Recently, a lot of CNN-based 2D pose estimation algorithms were proposed [48–55]. All these methods provide 2D body parts locations that can be used for 3D human pose estimation. For example, Cao et al. [48] managed to efficiently detect the 2D poses of multiple persons in an image using a nonparametric representation, which allows learning associations between body parts of everyone in the image. In [4] 2D body parts locations (detected by [56]) are used to automatically estimate the 3D pose and shape of the human body from a single unconstrained image. However, this method is not real-time and works for a single person only. [57] can recover any 3D articulated object given its joints correspondences but also works for a single object and assumes 2D joints positions to be given.

The methods for the real-time estimation of the 3D human pose with a monocular RGB camera are the most related to our algorithm. Only a few works target the problem for estimating temporally stable results which are directly usable in practical applications. The top-performing 3D pose estimation methods are based on CNNs [3, 12, 13, 30, 58, 59]. [13] used a 100-layer CNN architecture to predict 2D and 3D poses simultaneously. However, this algorithm is unsuitable for real-time applications due to the additional preprocessing steps such as bounding box extraction. [3] uses CNN to detect 2D and 3D joint's locations. Thereafter, an optimization-based skeletal fitting method is applied to estimate 3D poses in real-time. In [58] a coarse-to-fine volumetric approach for 3D human pose estimation from a single image is introduced. Their method is computationally expensive and does not support temporal stability. [60] is a new method that estimates the 3D pose of hands, face, and body from a single RGB image. However, all these methods are trained for single human tracking and do not generalize to multi-objects. On the other hand, we propose a multi-object 3D pose estimation approach that automatically estimates object-specific 3D skeleton and initial 3D location for each articulated object in the scene. Then, the 3D pose of every object is estimated by means of optimizing an energy function for skeleton fitting.

2. OVERVIEW

In the following sections, we focus on human motion tracking. An outline of our human motion tracking pipeline is illustrated in Fig. 3. However, the same steps can be applied to any articulated object with defined skeleton and relation between the lengths of its parts.

Input to our approach can be either the live stream of a monocular RGB camera (e.g. webcam), YouTube video, or video captured with a mobile phone camera. Any of these inputs yield a single frame I_i at discrete points in time $i = \{1, 2, 3, ...\}$. For frame I_i, the final output is $\mathbf{X} = \{X_1, ..., X_{prsn}\}$ where $prsn$ is the number of persons in the scene. X_j is the 3D skeletal pose parameters of the person with index j. This output is temporally consistent which makes it perfect for applications such as virtual reality and character control. Our algorithm works with any camera (i.e. webcam, static, moving, or spherical camera with strong distortion) and general scenes (e.g. outdoors with strong illumination changes).

Many related works [3, 7, 61] assume given person-specific 3D skeletons or initial pose parameters X_{init}. They also assume that the number of skeletons is fixed over the whole sequence. As these assumptions limits the benefit of the algorithm, we automatically estimate the number of persons in the scene. Then, we automatically generate person-specific 3D skeletons and estimate their initial 3D location. All these automatic steps are achieved in real-time before tracking each sequence which we refer to as the **initialization phase**. The basic idea of our automatic skeleton generation approach is to adapt a default human skeleton to the length of each bone of each person. To this end, anthropometric data tables are used to define the length of each bone as a function of the height of each person; see Section 4 for details.

To start the tracking process, it's not enough to generate the person-specific 3D skeletons, We also need to define the initial poses of each person. Existing human motion tracking algorithms either estimate the initial pose manually or use computationally expensive methods such as [4]. In this paper, we automatically estimate the 3D root location of each person in the scene which resolves this limitation; see Section 4 for details.

In the second phase (i.e. tracking phase), we start with a CNN-based approach [48, 50] to estimate the 2D body joints locations for each person in the scene. The output of this step is the matrix $\mathbf{J} = [J_1, ..., J_{prsn}]$ where J_i contains body-parts locations of person i. However, the order and number of the persons in $\mathbf{J}$ may vary from frame to frame. Therefore, we use Equation 4 to find the 2D body-parts positions J_i corresponding to each specific 3D skeleton. Thereafter, we dynamically generate 3D skeletons for persons who enter the scene and delete the skeletons of those who left; see Section 4 for details. The pose parameters $\mathbf{X} = \{X_1, ..., X_{prsn}\}$ are optimized given the 2D body-parts positions with the following energy function at each time frame I_i:

$$E(\mathbf{X}, \mathbf{J}) = E_{FIT}(\mathbf{X}, \mathbf{J}) - w_L E_L(\mathbf{X}) - w_A E_A(\mathbf{X}) \qquad (1)$$

where $E_{FIT}(\mathbf{X}, \mathbf{J})$ is the skeletons fitting term (Equation 5). $E_L(\mathbf{X})$ enforces predefined joint limits, and $E_A(\mathbf{X})$ is a smoothness term that prevents sudden change in the motion; see [5] for details. Once the optimal values of the weights $w_l = 0.1$and $w_a = 0.05$ is found, we fixed these weights in all experiments. The proposed energy function is smooth and analytically differentiable. Therefore, it can be optimized efficiently using standard gradient ascent initialized with the initial pose estimated in Section 4.

4. MULTI-PERSON 3D POSE ESTIMATION

In this section, we introduce a detail description of the components of our pipeline. In the first two subsections, we discuss the initialization phase. The tracking phase is explained in last subsection.

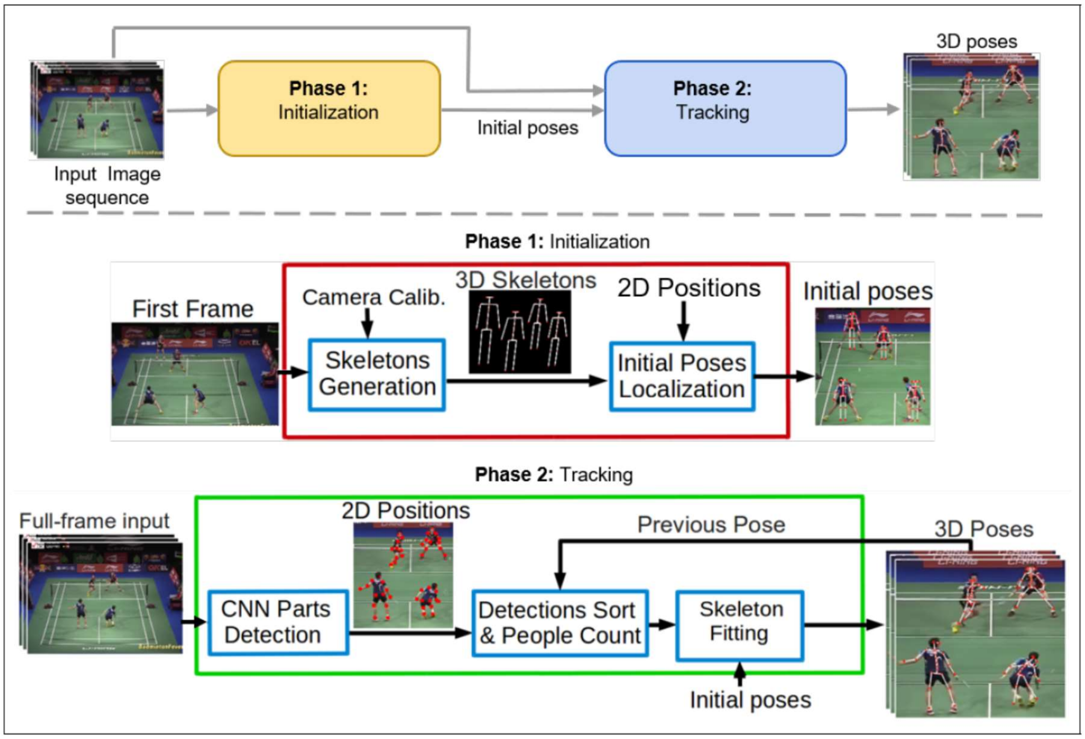


Fig 3. Overview of the proposed pipeline. In **Phase 1**, we generate multiple person-specific 3D skeletons and their initial location. Then, global 3D poses are estimated by fitting the skeletons to 2D body-parts positions. The **Detections Sort & People Count** step generates or deletes 3D skeletons for persons who enter or leave the scene.

4.1 3D Skeletons Generation

Human motion tracking algorithms require a 3D model with a properly personalized skeleton to successfully track a single person. Many related works consider model personalization as a different problem. Thus, they solve this task manually [6, 7], which greatly reduces their applicability for real applications. In this section, we propose an automatic approach that generates a skeleton specific to each person. Other algorithms [62] require many RGB cameras to automatically estimate the body mode. Therefore, we introduce a simple human 3D skeleton personalization approach that works with a monocular camera.

In this approach, we generate a default skeleton for every person. The initial number of persons is automatically estimated given the 2D detections of the first frame. Then, we adapt the bone length of each skeleton to match the corresponding person. Our default skeleton consists of 25 bones and 26 joints; see [6] for details. The anthropomorphic data tables [63] allow to define the length of each bone in the skeleton as a function of the height of the person. With these tables, the skeleton generation task is simplified to the estimation of the height of the person which can be estimated from a monocular RGB camera by back-projecting 2D features of an object into the 3D scene space; see [38, 64] for details. This allows personalizing the default skeletons to every person in the scene.

4.2 Skeletons Localization

The Skeleton generation step is very important. However, it is not enough to start the tracking process. To this end, we still need to estimate initial 3D pose of each person. Many motion tracking methods perform this step manually or with a different computationally expensive approach such as [4]. Fortunately, our algorithm is stable even with inaccurate initial poses. Therefore, we simplify the initial pose estimation problem to the estimation of the initial root position (i.e. 3D point between hips) of each person. To this end, we use the heights H_i^{3D} of each person i, their 2D body-part detections in the first frame J_i, and the monocular camera focal length f. The person heights H_i^{3D} and 2D body-parts detections J_i are estimated; as discussed in Section 4.1 using the CNN-based algorithm. As the upper body is usually more visible than the lower body, we use the height of the torso $H_{trs,i}^{3D} \approx 0.3 * H_i^{3D}$ for estimating the root depth. The 2D height of the torso $H_{trs,i}^{2D}$ is the distance between the neck $j_{nck,j}$ and the root $j_{rt,i} = (j_{lhip,i} + j_{rhip,i})/2$. With this, the depth of the root is calculated by:

$$z_i^{3D} = \frac{H_{trs,i}^{3D} * f}{H_{trs,i}^{2D}}. \tag{2}$$

Then, the 3D root position is calculated by:

$$\{x_i^{3D}, y_i^{3D}, z_i^{3D}\} = \mathbf{\Phi}^{-1}(j_{rt,i}^{x} * z_i^{3D}, j_{rt,i}^{y} * z_i^{3D}, z_i^{3D}) \tag{3}$$

where $\mathbf{\Phi}$ is the projection operator. Then, our algorithm automatically moves each skeleton such that its root position matches the root location of the corresponding person in 3D space.

4.3 Skeleton Fitting

In the first phase, we introduced methods for estimating personalized skeletons and their initial 3D locations. Once these estimates are available at the bigging of the tracking process, we can start the tracking phase which continues until the last frame in the image sequence. The first step of the tracking phase is the estimation of the 2D body-parts positions. Recently, many CNN based methods managed to accurately estimate these 2D body-parts positions [48–50]. Any of these methods can be used in our framework. However, we used both [48] and [50] in our experiments. As [48] achieves reasonable accuracy with multi-person, most of our results are based on this algorithm. Therefore, in this section, we assume, without loss of generality, that 2D body-part positions are estimated by [48].

As the 2D body-part detection algorithm does not have any temporal relation between consecutive frames, the order of the resulting 2D body-part detections in $\mathbf{J} = [J_1, ..., J_{prsn}]$ for one frame chance over time. It is important to solve this problem because the body-parts positions J_m may correspond to a different person in each frame. For this reason, the next step in our tracking phase is to associate each existing 3D skeleton with the corresponding 2D detections J_m in each frame. To this end, we define a similarity measure between the skeleton defined by pose parameters X_k and $J_m = [j_{m,1}, ...j_{m,prt}]$ where prt is the number of 2D body part detections of one person. This is done by first projecting the 3D joint positions defined by X_k into the 2D image plane using the projection operator $\mathbf{\Phi}$. Thereafter, the distance between each projected 3D joint and the corresponding 2D detection is calculated. The final similarity between skeleton with index k and detections in J_m is computed using the following equation:

$$SIM_{k,m} = \sum_{l=1}^{n_{prt}} \|\mathbf{\Phi}(\mathbf{f}_{k,l}(X_k)) - j_{m,l}\| \tag{4}$$

where $\mathbf{f}_{k,l}$ is the 3D joint position corresponding to the 2D body part $j_{m,l}$. This allows associating each skeleton with index k with the 2D detection $i = \arg\min_x SIM_{k,x}$.

Dynamic Number of Persons: For tracking varying number of persons, we need to generate a new 3D skeleton for each person who enters the scene and remove the skeleton of those who leave the scene. After associating each 3D skeleton with the corresponding 2D detections J_i, some items of $\mathbf{J}$ may be left without a corresponding 3D skeleton. These items correspond to either persons who just entered the scene or false positive detection of a human. To distinguish between these two cases, we use the confidence of each body part detection in J_i which is another output of the CNN-based approach. This confidence allows to compute a score for each J_i which corresponds to probability of a new person entering the scene. For each new J_i with score above the threshold $\alpha = 0.5$, we generate 3D skeleton for the corresponding person and estimate the respective initial 3D location. On the other hand, in case of a person leaving the scene or largely occluded, J_i corresponding to an existing skeleton will either have very low score or disappear from $\mathbf{J}$. In both cases, we remove that skeleton.

Our multi-person skeleton fitting term measures the similarities between a given skeleton pose *Xn* corresponding to one of the persons and 2D body-parts positions J_n of that person. Similar to Equation 4, we project each 3D joint position and calculate the distance to the corresponding 2D detection $j_{n,l}$. The final fitting term is defined as:

$$E_{FIT}(X,J) = \sum_{n=1}^{n_{prsn}} \sum_{l=1}^{n_{prt}} w(j_{n,l}) \exp\left(-\frac{\|\mathbf{\Phi}(\mathbf{f}_{n,l}(X_n)) - j_{n,l}\|^2}{\sigma^2}\right) \tag{5}$$

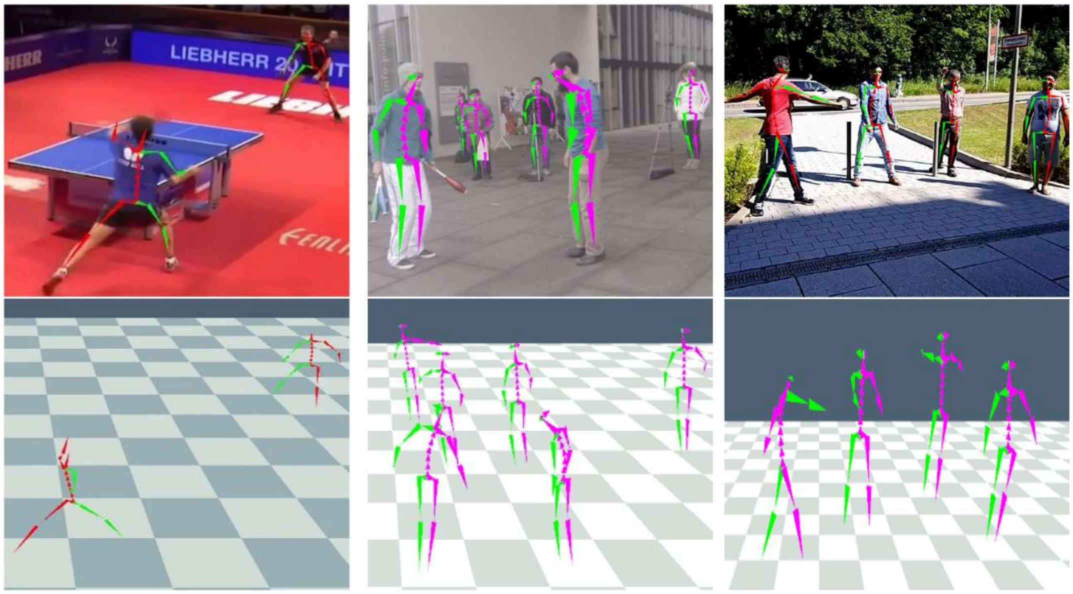
where $w(j_{n,l})$ is the confidence of the 2D body-parts detection $j_{n,l}$. This confidence is estimated by the CNN body-parts estimation method.

Temporal Smoothness: Applying per-frame pose estimation techniques on a video does not ensure temporal consistency of motion. Thus, small pose inaccuracies lead to temporal jitter. Therefore, we combine our multi-person skeletons fitting energy with temporal filtering and smoothing in a joint optimization framework to obtain an accurate, temporally stable and robust result; see Equation 1.

5. EXPERIMENTS

We prove the efficiency of the proposed algorithm through intensive experimental evaluations on more than 20 challenging real-world sequences. Some of these sequences are YouTube video which includes varying number of persons performing complex and fast motions. Other sequences were captured outdoors with mobile-phones and spherical cameras. To illustrate the applicability of our algorithm for real-time applications, we performed live tracking of multiple persons at around 23Hz with a low-quality webcam. Moreover, we tracked several sequences from public datasets such as the Human3.6M [34] and the Marconi [6]. These sequences vary in numbers and identities of persons, the lighting conditions, complexity and speed of the motion, camera types (e.g. mobile-phone, GoPro, spherical cameras, and webcams), the frame rates, and the frame resolutions. Our algorithm automatically generates 3D skeletons and estimates initial poses for multiple persons. It does not need bounding box cropping. Thus, our experimental setup is very simple. It produces high-quality motion tracking, given the input images and the focal length of a single RGB camera. The run-time of our algorithm depends on the number of objects in the scene, the complexity of the motion, and the resolution of the input frames. Our experiments are performed on an 8-core Xeon CPU and a GeForce GTX 1080 GPU. The average processing time of a single frame from a single-person sequence is 44 milliseconds. The 2D body parts detection [48] takes 32 milliseconds while the 3D skeleton fitting needs 12 milliseconds. The initialization phase takes around 0.01 milliseconds, given the body parts detections of the first frame and the height of each person. The run time can be improved by using a better GPU.

Our algorithm can use any 2D body-parts detection method. Therefore, we show results with two different body-parts detection methods. In **Implementation 1,** we use [48] for 2D body-parts detections. This implementation is discussed in detail in Section 4. [48] does not need cropping to track multi-person.



(a) Community videos **(b) Marconi dataset** **(d) Mobile phone video**

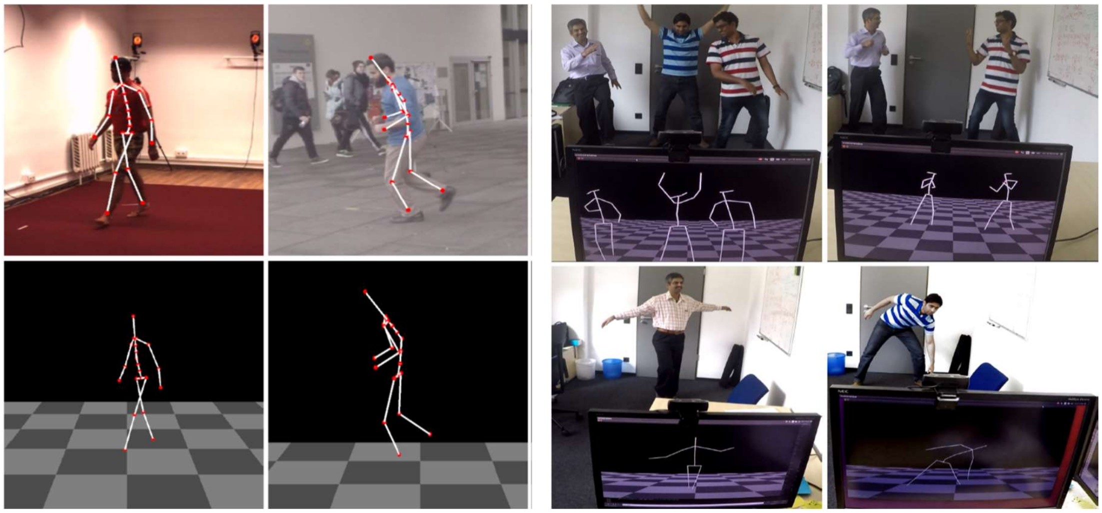
Fig 4. Sample results with overlaid 2D skeletons estimated with Implementation 1 (top) and respective 3D reconstructions (bottom) show successful multi-person tracking in challenging scenarios. (a) shows multi-person pose estimation results from YouTube videos. (b) shows results over selected difficult sequences from the Marconi dataset. (c) shows tracking results with strong illumination changes in an outdoor scene captured using a mobile phone camera.

Implementation 2 is based on [50] which requires cropping of every person. Fortunately, we can perform cropping automatically without significant change to the original pipeline in Fig. 3. To this end, we extrapolate the persons 3D poses of the previous frame to generate a rough estimate of the 3D poses of the current frame. Then, the bounding box of each person is estimated by projecting his 3D skeleton to the camera view. This allows to crop and scale each person. With this additional automatic step, any algorithms such as [50] can be used instead of [48] in our pipeline for 2D body part detections.

Qualitative Results: We used **Implementation 1** to track more than 15 sequences. Fig. 2 and Fig. 4 show samples from the tracked sequences. This illustrates the effectiveness of our algorithm for tracking sequences with many (i.e. up to eight) persons performing complex and fast motions under strong distortion and strong lighting variations. Previous monocular methods such as [3, 12, 13] fail to track these sequences in real-time. Fig. 5 shows the 3D pose reconstruction results based on **Implementation 2**. Two sequences from the public datasets the Human3.6M (Fig. 5(a)) and the Marconi (Fig. 5(b)) are successfully tracked.

To demonstrate the benefit of the proposed algorithm for real-time applications, we tracked the motion of multiple persons from a live stream of a webcam. Fig. 5(c) shows that our algorithm generates natural motion even with challenging scenarios. Furthermore, we track sequences with several people entering and leaving the scene. Our algorithm automatically detects the change in the scene and generates or deletes the corresponding 3D skeletons on the fly while tracking.

Comparison: Fig. 6(a) illustrates the comparison between the accuracy of our algorithm and the accuracy of [3, 65] on two sequences. Our algorithm accurately tracks the persons in the two sequences in real-time. On the other hand, [65] works only offline and [3] tracks only one of the two persons in the scene with low accuracy.



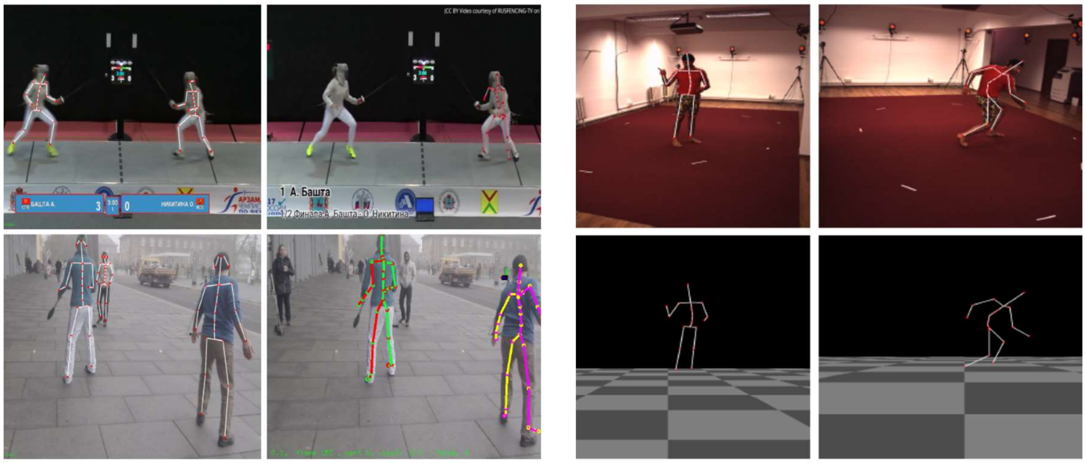
(a) H3.6M Sample **(b) Marconi Sample** **(c) Rea- time Tracking**

Fig 5. Sample images from the H3.6M dataset (a) and the Marconi dataset (b) with overlaid 2D Skeleton alongwith respective 3D pose recovery using **Implementation 2**. (c) The real-time 3D pose estimation from the stream of a webcam with **Implementation 1** (Top) and **Implementation 2** (Bottom).

Ablation Study: we present comprehensive ablation studies on various components of our algorithm by creating different alternatives pipelines. The first alternative pipeline is produced by removing the skeleton generation step. This means that the same default skeleton is used for every person without adaptation to his size. The second alternative pipeline is produced by disabling the initial pose localization step where the initial pose parameters are set to zero. We evaluated these alternatives by tracking the Walking sequence (i.e. Subject S9) from the Human3.6M dataset [34]. The Mean Per Joint Position Error (MPJPE) with our full algorithm is 90mm. The error increase with the first alternative pipeline to 460mm. However, the algorithm fails completely with the second alternative pipeline as the energy function is non-convex which leads to stuck in a local maxima; see Fig. 7.

Quantitative Evaluation: We quantitatively evaluate our algorithm using several sequences (i.e. the Directions, Posing, and Waiting) from the Human3.6M dataset which capture Subject S9. See Fig. 6(b) for sample 2D skeletons (i.e. overlaid on the input RGB images) and 3D skeletons reconstructions. The average error of all frames in these three sequences is 159.33 mm.

Discussion: Our approach is subject to a few limitations. The first limitation is the low accuracy of the depth estimation, especially in the case of wrists and ankles occlusion. This led to relatively higher 3D joint position errors. Nevertheless, this is a common problem in all monocular approaches as depth estimation is severely ill-posed. As a result of this fact, a slight inaccuracy in the 2D joints estimation leads to a large error in the depth estimation. Unlike other approaches, our algorithm can recover from tracking failures, even after long occlusion of many body-parts. The results of tracking many sequences illustrated that our algorithm succeeds in challenging multi-person scenarios where many other motion tracking methods fail. Furthermore, our algorithm achieved high temporal stability and reasonable accuracy. This accuracy can also be further improved by using 2D body part detectors that are more stable to occlusions.



(a) comparison with both [3] and [65] **(b) H3.6M Samples used in quan. evaluations**

Fig 6. (a) Side-by-side comparison of our method (left column) against Mehta et al. [3] (top of the second column) and the offline method of Elhayek et al. [65] (bottom of the second column). Our approach succeeds in accurately tracking all persons. (b) Sample images from H3.6M sequences used for quantitative evaluations. We show tracking results of Directions (third column) and Posing (fourth column) sequences for Subject S9 whose Mean Per Joint Position Error is 153mm and 158mm respectively.

5. Conclusion

We have presented a pipeline for tracking the instantaneous pose of multiple articulated objects from a single RGB video. To illustrate the ability of this pipeline, we showed competitive results for tracking human motion in real-time using a monocular RGB camera. Our pipeline leverages the accuracy of the discriminative and generative approaches. To this end, we utilize the discriminative deep learning-based methods for estimating the 2D joints positions of the articulated object. Then, the generative energy function is used to fit the 3D skeletons to these joint's positions. Our fully automatic algorithm estimates 3D kinematic poses of multiple objects in a temporally stable manner. It automatically detects the number of objects in the scene and generates corresponding object-specific 3D skeletons. It also automatically estimates the initial 3D location of each object which allows to define their coarse initial poses. The 2D joints detections can be estimated using any discriminative method which allows to easily upgrade our algorithm with any progress in 2D pose estimation. Our algorithm dynamically generates 3D skeletons for objects that enter the scene and delete the skeletons of those which leave. In contrast to previous works, our fully automatic algorithm can operate with multiple objects in real-time without the need of bounding boxes. This makes our algorithm optimal for many applications. Our intensive experiments demonstrated the effectiveness of our system for tracking sequences with strong distortion, strong illumination changes, and complex motions. Moreover, we tracked multiple humans in real-time using live streaming from a webcam. As future work, we are going to extend our algorithm for human pose and shape estimation. Furthermore, to improve the run-time, we will employ more advanced optimization algorithms.

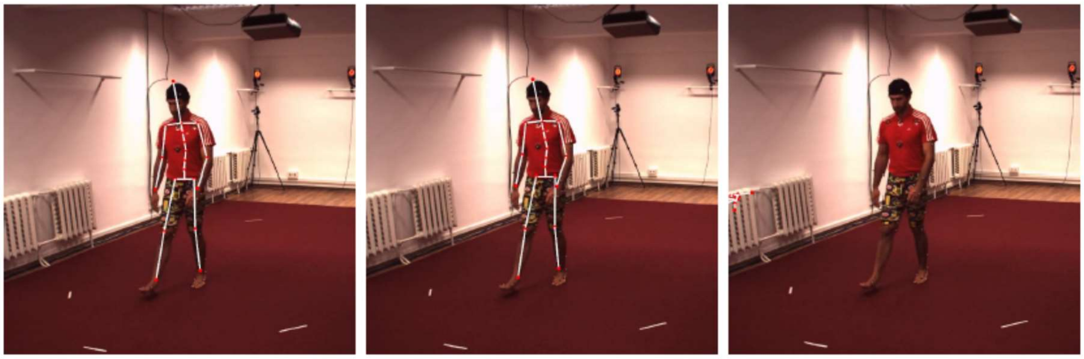


Fig 7. Ablation studies on our algorithm's components. **Left**: tracking result of our full pipeline; MPJPE 90mm. **Middle**: an alternative of our algorithm constructed by removing the skeleton generation step (i.e. using the default skeleton); MPJPE 460mm. **Right**: second alternative constructed by removing initial pose localization step which fails completely.

References

1. Kanazawa A, Black MJ, Jacobs DW, Malik J. End-to-End Recovery of Human Shape and Pose. In: Computer Vision and Pattern Recognition (CVPR); 2018. p. 7122–7131.

2. Rogez G, Weinzaepfel P, Schmid C. LCR-Net++: Multi-person 2D and 3D Pose Detection in Natural Images. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI). 2019.

3. Mehta D, Sridhar S, Sotnychenko O, Rhodin H, Shafiei M, Seidel HP, et al. VNect: Real-time 3D Human Pose Estimation with a Single RGB Camera. In: ACM Transactions on Graphics. vol. 36; 2017.

4. Bogo F, Kanazawa A, Lassner C, Gehler P, Romero J, Black MJ. Keep It SMPL: Automatic Estimation of 3D Human Pose and Shape from a Single Image. In: ECCV; 2016. p. 561–578.

5. Stoll C, Hasler N, Gall J, Seidel HP, Theobalt C. Fast Articulated Motion Tracking using a Sums of Gaussians Body Model. In: ICCV; 2011.

6. Elhayek A, Aguiar E, Jain A, Tompson J, Pishchulin L, Andriluka M, et al. Efficient ConvNet-based Marker-less Motion Capture in General Scenes with a Low Number of Cameras. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2015. 387

7. Elhayek A, Stoll C, Hasler N, Kim KI, Seidel HP, Theobaltl C. Spatio-temporal Motion Tracking with Unsynchronized Cameras. In: Proc. CVPR; 2012.

8. Elhayek A, Stoll C, Hasler N, Kim KI, Theobaltl C. Outdoor Human Motion Capture by Simultaneous Optimization of Pose and Camera Parameters. In: Proc. CGF; 2014.

9. Baak A, M¨uller M, Bharaj G, Seidel HP, Theobalt C. A Data-Driven Approach for Real-Time Full Body Pose Reconstruction from a Depth Camera. In: Proc. ICCV; 2011. p. 1092–1099. 394

10. Ye M, Shen Y, Du C, Pan Z, Yang R. Real-Time Simultaneous Pose and Shape Estimation for Articulated Objects Using a Single Depth Camera. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2016;38(8):1517–1532. doi:10.1109/TPAMI.2016.2557783.

11. Dou M, Khamis S, Degtyarev Y, Davidson P, Fanello SR, Kowdle A, et al. Fusion4d: Real-time performance capture of challenging scenes. ACM Transactions on Graphics (TOG). 2016;35(4):114.

12. Zhou X, Zhu M, Pavlakos G, Leonardos S, Derpanis KG, Daniilidis K. MonoCap: Monocular Human Motion Capture using a CNN Coupled with a Geometric Prior. CoRR. 2017;abs/1701.02354.

13. Mehta D, Rhodin H, Casas D, Sotnychenko O, Xu W, Theobalt C. Monocular 3D Human Pose Estimation Using Transfer Learning and Improved CNN Supervision. arXiv preprint arXiv:161109813. 2016.

14. Elhayek A, Kovalenko O, Murthy P, Malik J, Stricker D. Fully Automatic Multi-person Human Motion Capture for VR Applications. In: EuroVR; 2018.

15. Yuan S, Garcia-Hernando G, Stenger B, Moon G, Chang JY, Lee KM, et al. Depth-Based 3D Hand Pose Estimation: From Current Achievements to Future Goals. In: IEEE CVPR; 2018.

16. Supancic JS, Rogez G, Yang Y, Shotton J, Ramanan D. Depth-based hand pose estimation: data, methods, and

challenges. In: IEEE international conference on computer vision; 2015. p. 1868–1876.

17. Moeslund T, Hilton A, Kr¨uger V. A survey of advances in vision-based human motion capture and analysis. CVIU. 2006;104(2):90–126.

18. Poppe R. Vision-based human motion analysis: An overview. CVIU. 2007;108(1-2):4–18.

19. Sigal L, Balan A, Black M. HumanEva: Synchronized Video and Motion Capture Dataset and Baseline Algorithm for Evaluation of Articulated Human Motion. IJCV. 2010;87:4–27.

20. Bregler C, Malik J. Tracking People with Twists and Exponential Maps. In: CVPR; 1998. p. 8–15.

21. Bogo F, Romero J, Loper M, Black MJ. FAUST: Dataset and evaluation for 3D mesh registration. In: CVPR; 2014.

22. Starck J, Hilton A. Model-Based Multiple View Reconstruction of People. In: ICCV; 2003. p. 915– 922.

23. Gall J, Rosenhahn B, Brox T, Seidel HP. Optimization and Filtering for Human Motion Capture – A Multi-Layer Framework. IJCV. 2010;87:75–92.

24. Bo L, Sminchisescu C. Twin Gaussian Processes for Structured Prediction. IJCV. 2010;87:28–52.

25. Lee CS, Elgammal A. Coupled Visual and Kinematic Manifold Models for Tracking. IJCV. 2010;87:118–139.

26. Li R, Tian TP, Sclaroff S, Yang MH. 3D Human Motion Tracking with a Coordinated Mixture of Factor Analyzers. IJCV. 2010;87:170–190.

27. Urtasun R, Fleet DJ, Fua P. Temporal Motion Models for Monocular and Multiview 3D Human Body Tracking. Comput Vis Image Underst. 2006;104(2):157–177. doi:10.1016/j.cviu.2006.08.006.

28. Jain A, Thorm¨ahlen T, Seidel HP, Theobalt C. MovieReshape: Tracking and Reshaping of Humans in Videos. ACM Trans Graph (Proc SIGGRAPH Asia 2010). 2010;29(5).

29. Rogge L, Klose F, Stengel M, Eisemann M, Magnor M. Garment Replacement in Monocular Video Sequences. ACM Transactions on Graphics. 2014;34(1):6:1–6:10.

30. Omran M, Lassner C, Pons-Moll G, Gehler PV, Schiele B. Neural Body Fitting: Unifying Deep Learning and Model Based Human Pose and Shape Estimation. In: 3D Vision (3DV); 2018. p. 484–494.

31. Sun X, Shang J, Liang S, Wei Y. Compositional Human Pose Regression. In: International Conference on Computer Vision (ICCV); 2017. p. 2621–2630.

32. Agarwal A, Triggs B. Recovering 3D human pose from monocular images. IEEE transactions on pattern analysis and machine intelligence. 2006;28(1):44–58.

33. Kostrikov I, Gall J. Depth Sweep Regression Forests for Estimating 3D Human Pose from Images. In: BMVC. vol. 1; 2014. p. 5.

34. Ionescu C, Papava D, Olaru V, Sminchisescu C. Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. IEEE transactions on pattern analysis and machine intelligence. 2014;36(7):1325–1339.

35. Li S, Chan AB. 3d human pose estimation from monocular images with deep convolutional neural network. In: Asian Conference on Computer Vision. Springer; 2014. p. 332–347.

36. Li S, Zhang W, Chan AB. Maximum-margin structured learning with deep networks for 3d human pose estimation. In: Proceedings of the IEEE International Conference on Computer Vision; 2015. p. 2848–2856.

37. Tekin B, Rozantsev A, Lepetit V, Fua P. Direct prediction of 3d body poses from motion compensated sequences. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2016. p. 991–1000.

38. Du Y, Wong Y, Liu Y, Han F, Gui Y, Wang Z, et al. Marker-less 3D human motion capture with monocular image sequence and height-maps. In: European Conference on Computer Vision. Springer; 2016. p. 20–36.

39. Chen W, Wang H, Li Y, Su H, Wang Z, Tu C, et al. Synthesizing Training Images for Boosting Human 3D Pose Estimation. In: 3D Vision (3DV); 2016.

40. Martinez J, Hossain R, Romero J, Little JJ. A Simple Yet Effective Baseline for 3d Human Pose Estimation. In: International Conference on Computer Vision (ICCV); 2017. p. 2659–2668.

41. Moreno-Noguer F. 3D Human Pose Estimation from a Single Image via Distance Matrix Regression. In: Computer Vision and Pattern Recognition (CVPR); 2017. p. 1561–1570.

42. Hossain MRI, Little JJ. Exploiting Temporal Information for 3D Human Pose Estimation. In: European Conference on Computer Vision (ECCV); 2018. p. 69–86.

43. Lee HJ, Chen Z. Determination of 3D human body postures from a single view. Computer Vision, Graphics, and Image Processing. 1985;30(2):148–168.

44. Akhter I, Black MJ. Pose-conditioned joint angle limits for 3D human pose reconstruction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2015. p. 1446–1455.

45. Valmadre J, Lucey S. Deterministic 3d human pose estimation using rigid structure. Computer Vision–ECCV 2010. 2010; p. 467–480.

46. Leonardos S, Zhou X, Daniilidis K. Articulated motion estimation from a monocular image sequence using spherical tangent bundles. In: Robotics and Automation (ICRA), 2016 IEEE International Conference on. IEEE; 2016. p. 587–593.

47. Fan X, Zheng K, Zhou Y, Wang S. Pose locality constrained representation for 3d human pose reconstruction. In: European Conference on Computer Vision. Springer; 2014. p. 174–188.

48. Cao Z, Simon T, Wei SE, Sheikh Y. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. In: CVPR; 2017.

49. Insafutdinov E, Pishchulin L, Andres B, Andriluka M, Schieke B. DeeperCut: A Deeper, Stronger, and Faster Multi-Person Pose Estimation Model. In: European Conference on Computer Vision (ECCV); 2016.Available from: http://arxiv.org/abs/1605.03170. 495

50. Bulat A, Tzimiropoulos eB Georgios", Matas J, Sebe N, Welling M. In: Human Pose Estimation via Convolutional Part Heatmap Regression. Cham: Springer International Publishing; 2016. p. 717–732.

51. Wei SE, Ramakrishna V, Kanade T, Sheikh Y. Convolutional pose machines. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2016. p. 4724–4732.

52. Toshev A, Szegedy C. Deeppose: Human pose estimation via deep neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2014. p. 1653–1660. 504

53. Charles J, Pfister T, Magee DR, Hogg DC, Zisserman A. Personalizing Human Video Pose Estimation. CoRR. 2015;abs/1511.06676.

54. Li J, Wang C, Zhu H, Mao Y, Fang HS, Lu C. CrowdPose: Efficient Crowded Scenes Pose Estimation and A New Benchmark. arXiv preprint arXiv:181200324. 2018.

55. Moon G, Chang J, Lee KM. PoseFix: Model-agnostic General Human Pose Refinement Network. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2019.

56. Pishchulin L, Insafutdinov E, Tang S, Andres B, Andriluka M, Gehler PV, et al. Deepcut: Joint subset partition and labeling for multi person pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2016. p. 4929–4937.

57. Kovalenko O, Golyanik V, Malik J, Elhayek A, Stricker D. Structure from Articulated Motion: An Accurate and Stable Monocular 3D Reconstruction Approach without Training Data. CoRR. 2019.

58. Pavlakos G, Zhou X, Derpanis KG, Daniilidis K. Coarse-to-Fine Volumetric Prediction for Single-Image 3D Human Pose. CoRR. 2016;abs/1611.07828.

59. LI S, Liu ZQ, Chan AB. Heterogeneous Multi-task Learning for Human Pose Estimation with Deep Convolutional Neural Network. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops; 2014.

60. Pavlakos G, Choutas V, Ghorbani N, Bolkart T, Osman AAA, Tzionas D, et al. Expressive Body Capture: 3D Hands, Face, and Body from a Single Image. In: Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR); 2019.

61. Elhayek A, Aguiar E, Jain A, Tompson J, Pishchulin L, Andriluka M, et al. Efficient ConvNet-based Marker-less Motion Capture in General Scenes with a Low Number of Cameras. In: Proc. CVPR; 2015.

62. Rhodin H, Robertini N, Casas D, Richardt C, Seidel HP, Theobalt C. General automatic human shape and motion capture using volumetric contour cues. In: European Conference on Computer Vision. Springer; 2016. p. 509–526.

63. C Gordon MM C Blackwell, Kristensen S. 2012 Anthropometric Survey of U.S. Army Personnel: Methods

and Summary Statistics. 2014;(Natick/TR-15/007).

64. Park SW, eun Kim T, Choi JS. Robust Estimation of Heights of Moving People Using a Single Camera. In: ICITCS; 2011.

65. Elhayek A, de Aguiar E, Jain A, Thompson J, Pishchulin L, Andriluka M, et al. MARCOnI: ConvNet-Based MARker-Less Motion Capture in Outdoor and Indoor Scenes. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2017;39(3):501–514. doi:10.1109/TPAMI.2016.2557779.

66. Andriluka M, Pishchulin L, Gehler P, Schiele B. 2D Human Pose Estimation: New Benchmark and State of the Art Analysis. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2014.