# A Hybrid Method for 3D Pose Estimation of Personalized Human Body Models

Ammar Qammaz[1,2], Damien Michel[2], Antonis Argyros[1,2]
[1]Computer Science Department, University of Crete, Greece
[2]Institute of Computer Science, FORTH, Greece
{ammarkov,michel,argyros}@ics.forth.gr

## Abstract

*We propose a new hybrid method for 3D human body pose estimation based on RGBD data. We treat this as an optimization problem that is solved using a stochastic optimization technique. The solution to the optimization problem is the pose parameters of a human model that register it to the available observations. Our method can make use of any skinned, articulated human body model. However, we focus on personalized models that can be acquired easily and automatically based on existing human scanning and mesh rigging techniques. Observations consist of the 3D structure of the human (measured by the RGBD camera) and the body joints locations (computed based on a discriminative, CNN-based component). A series of quantitative and qualitative experiments demonstrate the accuracy and the benefits of the proposed approach. In particular, we show that the proposed approach achieves state of the art results compared to competitive methods and that the use of personalized body models improve significantly the accuracy in 3D human pose estimation.*

## 1. Introduction

Due to its theoretical importance and practical significance, vision-based human motion capture has attracted a lot of research and development efforts. Most commercial solutions make use of special markers. Evidently, markerless motion capture techniques are much preferable because of their unobtrusiveness and their lower cost and complexity. Despite the great progress in the field, accurate, robust and efficient 3D human motion capture from markerless visual input in unconstrained settings is not yet possible.

Markerless human motion capture techniques may be classified into three broad classes, the bottom-up *discriminative* methods, the top-down *generative* methods and the hybrid ones. Generative methods can be very accurate, provide physically plausible solutions and do not require training. However, typically, they are computationally demanding, require initialization and can suffer from drift and track loss. Discriminative methods perform single frame pose estimation, do not require initialization and are computationally efficient. On the other hand, they rely on big collections of annotated training data and their solution is not always physically plausible. Hybrid methods integrate elements from discriminative and generative methods in an effort to combine the merits of both worlds.

In this paper, we propose a new hybrid approach. 3D human pose estimation is the result of the optimization of an objective function consisting of two terms. The first term quantifies the discrepancy between the 3D structure of a rendered human body model and the actual, observed one. The second term quantifies the displacement of the human body joints as estimated by a CNN-based discriminative component [3, 46] and the corresponding joints of the hypothesized 3D model. Several existing hybrid methods have *loosely coupled* generative and discriminative components. Typically, in each frame, the discriminative part provides an estimation of 3D human pose which is then refined by the generative component. On the contrary, in this paper we propose *tight integration*, in the sense that information resulting from the discriminative component is directly used in the optimization loop of the generative one. As it is shown experimentally, this tight integration and the proper balancing of the two terms achieves significantly better results than any of the terms alone. The joints localization term facilitates automatic initialization and recovery from drifts. The depth term safeguards from inaccuracies or missing information of the discriminative component. Moreover, it enforces pose solutions that are consistent with a body whose shape parameters do not vary in time.

Our method can employ any skinned, articulated human body model. However, we focus on personalized models that can be constructed automatically, in a couple of minutes, based on a registration procedure [36] coupled with a state of the art human avatar rigging method [10]. We show experimentally that by using such models the accuracy in tracking can be improved over using generic human body models.

All findings are supported by quantitative and qualitative

experimental results on datasets annotated with ground truth and in comparison with competitive methods.

In summary, the contributions of this work are:

- We propose a novel, hybrid method that performs single frame 3D human body pose estimation by integrating *tightly* a generative and a discriminative component. The method provides physically plausible solutions, does not require training, does not require initialization, outperforms competitive methods in pose estimation accuracy and operates at 9 Hz.

- We show that by employing easy-to-acquire body models, pose estimation accuracy is considerably increased compared to using generic body models and compared to existing methods.

- We make available a new dataset that contains ground truth on several human body models and 3D motions.

### 1.1. Related work

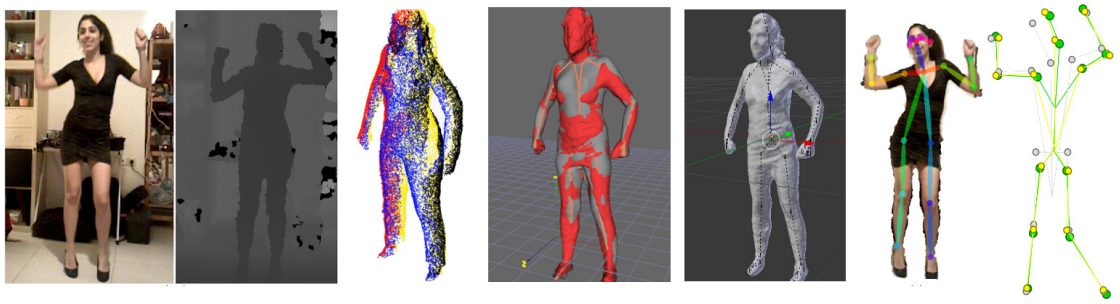
Works such as [25, 32] and [5] provide early surveys for conventional and depth cameras, respectively. As reported in [23], discriminative human pose estimation methods [39, 2, 37, 31, 38, 41] map a set of extracted image features to the human pose space. This is achieved through training over a large database of known poses. A variety of methods is defined based on the employed features, the mapping method and the actual training poses database. Recent approaches based on CNNs have produced very promising results [34, 48, 19, 20, 35, 3, 46]. A very recent work is VNect [20] which uses a convolutional neural network to acquire a 2D pose from an RGB frame and then performs regression using a kinematic skeleton to estimate 3D joints from the data. Although performance is real-time and output quality is very good, the lack of detailed knowledge about the tracked model and the absence of depth information have a negative impact on accuracy. The LCR-net [35] is another detection plus regression framework which converts 2D proposals from RGB images to 3D joints but lacks the high quality model and 3D rendering capabilities that are permitted when an RGBD sensor is used. Discriminative methods perform single frame pose estimation, so they don't rely on temporal continuity. Thus, they do not require initialization and they don't suffer from drift. Their offline training is computationally demanding, while their online runtime is rather good.

Generative approaches [11, 8, 9, 33, 13, 12, 44, 7, 21, 50] use a model of the human body and estimate its position, orientation and joint angles that bring the appearance of this model in accordance to the visual input. The model is usually made of a skeleton and an attached surface, which in some cases [13] is allowed to deform. Instead of estimating the full body model in a single step, a variety of methods first identify body parts. Then, they either report them as the final solution or they further assemble them into a full model [37, 38]. Generative methods rely on an objective function that quantifies the discrepancy between a model pose hypothesis and the actual visual input. The minimization of the objective function over the possible poses, determines the one that best explains the available observations. This amounts to the exploration of the high dimensional space of human poses. The size of the search space can be reduced by employing kinematic constrains based on biomechanical data that exclude non realistic poses. Further reductions can be achieved by constraining also the dynamics, i.e., by employing Kalman filters [24]. However, this requires learning of the dynamics of specific human motions and thus reduces the generality of the approach. Another way to deal with the high dimensionality of the search space is to perform local searches in the vicinity of the solution of the previous frame. This works fine under the assumption of human motion with temporal continuity. However, the violation of this assumption may cause drift and track loss. Local search also means that tracking needs to be initialized for the first frame. Due to their generative nature, the computational cost of the online process is typically high. On the other hand, the employed model can be changed easily, and the whole search space can be explored without the requirement for offline training.

Hybrid methods that integrate discriminative and generative components have been proposed [16, 1, 14, 30, 47, 49, 22] to combine the benefits of both worlds. Hybrid methods achieve the accuracy of the generative ones without need for initialization and with robustness to tracking failures. The method proposed in this work falls in this category of human pose estimation methods.

## 2. Pose estimation of personalized body models

The proposed method is summarized in Fig. 1. An RGBD frame is denoted as $o = (c^o, d^o)$, where $c^o$ and $d^o$ stand for the RGB and depth frames, respectively. The proposed method capitalizes on a parametric skinned model of the human body (Section 2.1). Human pose estimation in a frame amounts to estimating the parameters of the model that is most compatible with visual observations. The discrepancy between the model and the observations is quantified by an objective function (Section 2.3) that has two terms. The first (Section 2.3.1), compares the 3D structure of the observed human with the 3D structure of the rendered model. The second (Section 2.3.2) compares the locations of the joints as they were estimated by a neural network (Section 2.2) to the locations of the joints of the model hypothesis. The optimization of the defined objective function is performed effectively and efficiently using Particle Swarm Optimization [6] (Section 2.4).



(a) (b) (c) (d)

Figure 1. Overview of our approach for 3D human pose estimation. (a) The input is a sequence of RGBD frames. (b) We leverage on personalized, parametric, skinned human body models that can be acquired easily [10]. (c) We also employ state of the art estimation of 2D human body joints [4]. We define a hybrid approach for fitting the model (b) in the observations (a) given the hints on the 2D joints (c). The result of this process is visualized in (d). The green skeleton is the ground truth, the gray is the neural network suggestion and the yellow is the result of the proposed approach. Although the neural network estimation is far from the ground truth, the proposed method manages to estimate accurately the 3D human pose because of the high quality model and the utilization of depth information.

### 2.1. Human body model

The proposed method can operate with a human body model that consists of a set of appropriately assembled geometric primitives as in [22], or with a skinned model. In this work, we are particularly interested in personalized, automatically acquired human body models. The acquisition of such a model $\mathbf{H}$ is performed in two steps (a) human body scanning and (b) model rigging. Body scanning is performed as described in [36]. Model acquisition requires only that the human subject stands with the T-pose in front of an RGBD camera, in 4 different orientations. These views are then registered automatically[1] to form a reconstruction of the human body. Articulating the scanned model through model rigging is performed automatically[2] as described in [10]. From a practical point of view, given the mentioned automated tools, the acquisition of a body model $\mathbf{H}$ can be performed in less than 2 min.

A body model $\mathbf{H}$ acquired this way has a total of 94 bones which also account for parts of the face, hand fingers *etc*. For the large-scale body tracking scenario we are interested in, we restrict ourselves to a subset of those body parts and the respective joints as listed in Table 1.

The 3D position of $\mathbf{H}$ is represented with three parameters. Four more parameters encode a quaternion-based representation of its global orientation. Joints are represented with roll/pitch/yaw angles. Thus, a pose of $\mathbf{H}$ is represented as a $3 + 4 + 29 = 36$D parameter vector $h$. The limits of 29 of these parameters are shown in Table 1. Setting these limits excludes several physically implausible poses. However, certain combinations of valid (i.e., within limits) joint angles still result in impossible body configurations.

Given an instantiation $h$ of the model $\mathbf{H}$ and camera calibration parameters, we can render $\mathbf{H}$ to the view of the camera, obtaining color and depth maps $r = (c^h, d^h)$ that are comparable to the observations.

| *Joint* | $roll_{min}$ | $roll_{max}$ | $pitch_{min}$ | $pitch_{max}$ | $yaw_{min}$ | $yaw_{max}$ |
|---|---|---|---|---|---|---|
| Neck | -0.4 | 0.4 | -0.3 | 0.3 | - | - |
| Spine | -0.5 | 0.5 | -0.2 | 0.5 | -1.5 | 1.5 |
| LC | -0.1 | 0.1 | - | - | -0.15 | 0.1 |
| RC | -0.1 | 0.1 | - | - | -0.1 | 0.15 |
| LS | -4 | 4 | -1.8 | 1.57 | -2 | 1.4 |
| RS | -4 | 4 | -1.8 | 1.57 | -1.4 | 2 |
| LE | - | - | - | - | -2.5 | 0 |
| RE | - | - | - | - | 0 | 2.5 |
| LH | -1.57 | 1.57 | -0.78 | 2 | -1.2 | 0.5 |
| RH | -1.57 | 1.57 | -0.78 | 2 | -0.5 | 1.2 |
| LK | - | - | -2.8 | 0.1 | - | - |
| RK | - | - | -2.8 | 0.1 | - | - |
| LA | - | - | -0.6 | -0.6 | -0.7 | 0.7 |
| RA | - | - | -0.6 | -0.6 | -0.7 | 0.7 |

Table 1. Limits (in radians) of joint angles of the human body model $\mathbf{H}$. Besides Neck and Spine, the joints are coded with two letters. The first (L, R) stands for Left/Right and the second (C, S, E, H, K, A) for Collar, Shoulder, Elbow, Hip, Knee and Ankle.

### 2.2. Localizing body joints

Given a color frame $c^o$, we employ the OpenPose neural network [3, 46] which computes the 2D locations $j^e$ of human body joints. Any source of 2D/3D joint locations can be used, however OpenPose has been used because of its accuracy and robustness. We expect the 2D estimations of the joints not to be perfectly accurate but we also expect the detected joints to have some consistency in the temporal do-

[1]Human body reconstruction software: https://goo.gl/jFFj6a
[2]Human body model rigging software: https://goo.gl/AEtX96

main and in relation to each other. Given $j^e$, we can sample the depth information $d^o$ to get a coarse estimate of the 3D locations $J^e$ of the joints. Inaccuracies come from the fact that these 3D points lie on the surface of the body, while joints do not. Such problems are even more pronounced for occluded joints, i.e., for the back shoulder in a side view of a body or in cases of crossed arms or legs. Despite such inaccuracies, a coarse estimation of the 3D locations $J^e$ proves useful during optimization.

## 2.3. Objective function

An objective function $E(h, o)$ has been designed to quantify the discrepancy between a model hypothesis $h$ and the actual observations $o$. Estimating the human pose at a certan frame amounts to finding the model parameters $h^*$ of $\mathbf{H}$ that minimize $E(h, o)$. In notation,

$$h^* \triangleq \arg\min_h E\left(h, o\right). \tag{1}$$

$E(h, o)$ consists of two terms, $E_D$ and $E_J$. Specifically,

$$E\left(h, o\right) = w_D E_D\left(h, d^o\right) + w_J E_J\left(h, o\right). \tag{2}$$

The first term measures the discrepancy between the observed depth map and the depth map resulting from the rendering of $\mathbf{H}$ according to $h$. The second term measures the displacement between the locations (both 2D and 3D) of the human body joints. The first term is weighted by a constant $w_D$ whose value is determined experimentally in Section 3.3. The weight $w_J$ is set to 1.

### 2.3.1 The depth term $E_D$

For a given hypothesis $h$, we render $\mathbf{H}$ to obtain a color image $c^h$ and a depth map $d^h$. $d^h$ is comparable to the actual, observed depth map $d^o$ and their similarity is a strong indication for a correct hypothesis $h$. This motivates the following definition of the error term $E_D$:

$$E_D\left(h, d^o\right) = \frac{1}{N_P} \sum_{p \in B} C\left(|d_p^h - d_p^o|, T\right). \tag{3}$$

In Eq.(3), $E_D$ sums the absolute depth differences $|d_p^h - d_p^o|$ for all points $p$ that belong to a bounding box $B$ containing the human figure. The clamping function $C(x, T)$ returns $x$ if $x \leq T$ and $T$ otherwise. This is used to robustify the error term and prevent spurious points/outliers from affecting it too much. In our implementation we set $T = 30$cm.

In RGBD camera depth readings, a value of 0 represents lack of measurement due to e.g., reflective materials or infrared interference. Such points are ignored in Eq.(3).

The normalization term $N_p$ requires special attention. Setting $N_p$ equal to the number of rendered model points is a bad choice, because this promotes hypotheses that project to only a few pixels in the image, instead of hypotheses that explain all the available observations. To avoid this, $N_p$ is set equal to the number of points inside $B$ that are close to the depth profile of the previous solution. This way, all hypotheses for a certain frame share the same normalization.

### 2.3.2 The joints location term $E_J$

As discussed in Section 2.2, OpenPose provides estimations $j^e$ of the 2D locations of human joints, which can then be lifted to 3D estimations $J^e$ by exploiting the depth information $d^o$. Additionally, given a hypothesis $h$ of $\mathbf{H}$, we can estimate the 3D joint locations noted as $J^h$ and also render them in the camera view to obtain $j^h$. Thus, we define an error term $E_{J2D}$ that penalizes discrepancies between the hypothesized ($j^h$) and estimated ($j^e$) 2D locations of joints:

$$E_{J2D}\left(h, c^o\right) = \frac{1}{W} \sum_{i=1}^{n_J} w_i ||j_i^h - j_i^e||_2. \tag{4}$$

Similarly, we define a term $E_{J3D}$ that penalizes discrepancies between the hypothesized ($J^h$) and estimated ($J^e$) 3D locations of joints:

$$E_{J3D}\left(h, o\right) = \frac{1}{W} \sum_{i=1}^{n_J} w_i ||J_i^h - J_i^e||_2. \tag{5}$$

Some types of joints are localized by OpenPose more accurately than others. As an example, the head and the knees are more accurately localized compared to hips. In order to compensate for this behavior of the detector, we employ a weighting scheme in both the 2D (Eq.(4)) and 3D (Eq.(5)) terms of the objective function that prioritizes more accurate joints. We use $w_i = 0.2$ for hips and $w_i = 1.8$ for ankles and wrists. In Eqs.(4) and (5), $W = \sum_{i=1}^{n_j} w_i$, where $n_J$ is the number of all considered joints.

The aggregated joints location term $E_J$ is defined as

$$E_J\left(h, o\right) = \alpha E_{J2D}\left(h, c^o\right) + (1 - \alpha) E_{J3D}\left(h, o\right), \tag{6}$$

where $\alpha = 0.97$ is a constant that balances the contributions of the 2D and 3D error terms and which was set experimentally (Section 3.3). It appears that this value of $\alpha$ gives a dominant role to $E_{J2D}$. However, this is not the case, as $\alpha$ needs to compensate also for the different arithmetic scales in which $E_{J2D}$ and $E_{J3D}$ are measured.

## 2.4. Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) [6] is a stochastic method that performs optimization by iteratively improving a candidate solution with respect to an error term characterizing its quality (objective function). PSO has been applied successfully to a number of vision problems such as object detection [40], head pose estimation [29], 3D hand

tracking [26, 27], 3D tracking of hands in interaction with objects [18] as well as 3D human pose tracking [44, 22]. The popularity of PSO as an optimization strategy for articulated motion tracking comes from its ability to handle large search spaces and noisy, multi-modal, non-differentiateable objective functions. Moreover, as shown in Section 3.4, PSO is ideal for parallel implementation on modern GPU architectures, permitting interactive framerates and a 100× speedup compared to the serial implementation.

PSO maintains a population of candidate solutions, called particles, that have a position $p$ and a velocity $V$ in the search space. The movement of each particle $p_i$ is influenced by the best position $P_i$ this particle has ever visited up to the current iteration/generation, and simultaneously guided towards the globally best known position $G$ in the search space (i.e., the best of all $P_i$s). Both these positions are updated as better ones are found by other particles. The update to the $k$-th generation is described by:

$$V_{i,k} = r_1 c_1 \left(P_i - p_{i,k-1}\right) + r_2 c_2 \left(G - p_{i,k-1}\right) + \omega V_{i,k-1} \tag{7}$$

$$p_{i,k} = p_{i,k-1} + V_{i,k}, \tag{8}$$

where $p_{i,k}$ and $V_{i,k}$, respectively, denote the position and velocity of the particle $p_i$ at the $k$-th generation, $r_i$ are samples of the uniform distribution $U(0,1)$, and $c_1$, $c_2$ and $\omega$ are parameters controlling the convergence speed of PSO. The particles are allowed to move within predefined ranges along each dimension of the search space (in our problem, these ranges are shown in Table 1). To enforce this constraint whenever it is violated, the respective velocity $V_{i,k}$ is reduced up to the point that the constraint is again satisfied. These steps are followed iteratively, until a fixed upper bound of generations is reached. Parameters $c_1$, $c_2$ and $w$, are set as proposed in [6], that is, $c_1 = 2.8$, $c_2 = 1.3$ and $\omega = 2/\left|2 - \psi - \sqrt{\psi^2 - 4\psi}\right|$, where $\psi = c_1 + c_2$.

For our 3D human pose estimation problem, PSO is used to minimize the objective function of Eq.(2) over candidate solutions $h$. For each incoming frame, particles are initialized around the solution for the previous frame. The space around that solution is made large enough to include the $J^e$ estimation for the current frame. This, together with the $E_J$ term in the objective function, are the elements that permit to the method to perform without initialization and to recover from potential tracking drifts. Perturbations [26] are used for particles during the optimization procedure in order to help them escape local minima towards the global best solution. We also keep a history of detections and the last 5 estimated poses are also considered as particles in every new frame to help us recover faster from low quality OpenPose estimations that may cause a momentary drift. In order to evaluate fairly the solution proposed in this paper, no motion prediction model is used to initialize PSO particles and no smoothing is being performed in the sequence of results. However, such techniques, are expected to improve pose estimation accuracy when employed.

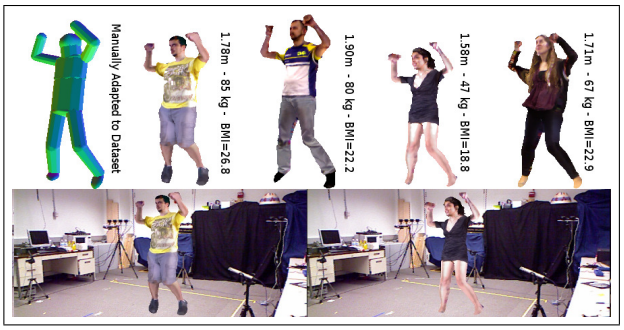



Figure 2. Snapshots of sample models used in the experiments. Top row: The primitives-based model **P** that was designed by hand and the four personalized models that were acquired automatically based on [36]. Bottom row: sample RGB frames from the rendering of $M_1$ and $F_1$ in the MHAD [43] dataset background, based on the CMU [42] mocap data.

## 3. Quantitative and qualitative assessment

The proposed method has been evaluated quantitatively on a synthetic data set (Section 3.1) annotated with ground truth. A first set of experiments investigated the effect of various parameters and design choices in the accuracy of the proposed method. In another set of experiments we compared the obtained performance to that of two baseline methods [22, 28]. The experimental evaluation is concluded with indicative qualitative results in RGBD sequences.

### 3.1. Synthetic dataset

The quantitative evaluation of our method requires a dataset containing RGBD frames of moving humans, together with ground truth regarding their 3D motion as well as their personalized 3D skinned model. A challenge we faced was that, to the best of our knowledge, there is no public dataset that features a complete, personalized skinned model of the actor, RGB+D information and 3D motion ground truth. For example, the Berkeley Multi modal Human Action Database (MHAD) [43] contains RGBD data and ground truth motion information, but no detailed skinned models of the actors. The CMU datasets [42] contain more challenging motions than MHAD, but contains no actual RGB information. Finally, the TNT15 dataset [45] lacks depth information. Moreover, the models therein are laser-scanned, therefore are much more accurate and noise-free compared to the ones we employ.

Thus, we constructed our own, synthetic dataset as follows. We scanned four different subjects using [36], two male ($\mathbf{M_1}$, $\mathbf{M_2}$) and two female ($\mathbf{F_1}$, $\mathbf{F_2}$). The four subjects differ significantly with respect to their sizes (height, weight, Body Mass Index). We also employed a primitives-

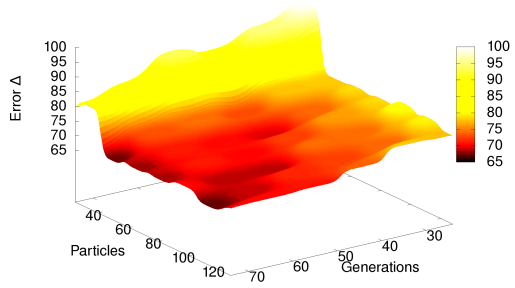



Figure 3. The error $\Delta$ as a function of the number of particles and generations in PSO optimization. See text for details.

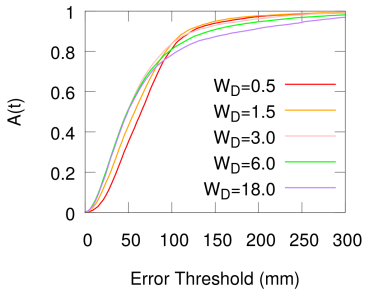



Figure 4. Tracking accuracy $A(t)$ for different weights $w_D$ of the depth term $E_D$ in the objective function.

based model ($\mathbf{P}$) whose dimensions can be adapted to best approximate a given human model. These models are illustrated in Fig. 2 (top). We then collected motion capture data from the CMU datasets [42], including a variety of motions like bending, jumping jacks, simultaneous twisting of torso and limbs, etc. Finally, we rendered the $\mathbf{M_i}$ and $\mathbf{F_i}$ models in the laboratory environment of the MHAD [43] datasets. Dual quaternion blending [17] was used to realize the skin deformations of the $\mathbf{M_i}$ and $\mathbf{F_i}$ models to avoid the "candy-wrapping" artifacts produced by standard linear blending. Thus, we obtained RGBD frames of known human models performing known, complex motions in a realistic environment. The final result[3] is four RGBD sequences of 720 frames each, of the same motions, performed by two male and two female subjects. We refer to these sequences as $MS_i$ and $FS_i$, respectively ($i \in \{1, 2\}$).

### 3.2. Evaluation metrics

To quantify the error in body pose estimation, we adopt the metric used in [15] which involves the Euclidean distances of skeleton joints in the ground truth and the corresponding points in the estimated body model. The average of all these distances over all the frames of the sequence constitutes the resulting error estimate $\Delta$.

Another metric reports the percentage $A(t)$ of these distances that are within a distance $t$ from their true location. We will refer to this metric as pose estimation accuracy. For example, an accuracy of $A(80) = 70\%$ for a sequence means that in all frames of the sequence, 70% of the joints were estimated within 80mm from the ground truth.

### 3.3. Quantitative results

**Determining the PSO budget:** PSO optimizes its objective function by evolving $p$ particles in $g$ generations (see Section 2.4). The proper selection of $p$ and $g$ is crucial because it influences the accuracy and the computational requirements of the method. We set $p$, $g$ based on the following experiment. We tracked the $MS_1$ sequence for all combinations of $p \in [40, 120]$ with a step of 10 and $g \in [30, 70]$ with a step of 10. For each particles/generations combination, we measured and averaged the error $\Delta$ (see Section 3.2) in 5 runs. Figure 3 shows $\Delta$ as a function of $p$ and $g$. It can be verified that a budget of 64 particles and 64 generations balances the error/computational resources tradeoff. Therefore, in all subsequent experiments we set $p = g = 64$.

**Tuning the objective function:** We performed experiments to investigate the influence of the weights $w_D$ and $w_J$ in the objective function of Eq.(2) that control the relative contribution of the depth and the joints terms. We investigated different values of $w_D$, maintaining $w_J = 1$. Figure 4 shows $A(t)$ for values $w_D \in \{0.5, 1.5, 3.0, 6.0, 18.0\}$. For a very broad range of errors $t$, $w_D = 3.0$ achieves, overall, the best $A(t)$. Around 10 cm, the plots exhibit a switch point, i.e., lower weights become preferable. This is attributed to the fact that for larger allowed errors, the significance of the depth term is less pronounced. With similar experiments, we determined experimentally the values of $\alpha = 0.97$ (Eq.(6)) as well as the values for the weights $w_i$ (Eqs.(4) and (5)).

**Proposed vs depth-only vs joints-only:** In another experiment, we compared the performance of the proposed method ($w_D = 3.0$, $w_J = 1.0$) with the case where the joints localization term $E_J$ is ignored ($w_J = 0.0$, $w_D = 1.0$) and the case where the depth term $E_D$ is ignored ($w_J = 1.0$, $w_D = 0.0$). Figure 5 illustrates the accuracy $A(t)$ obtained in the three cases for the $MS_1$ (left) and the $FS_1$ (right) sequences. It can be verified that the depth alone performs the worst. Optimization only with the OpenPose proposals performs better than optimization only with depth. However, fusing and balancing the two terms achieves better performance than any of the terms alone.

Table 2 shows the error $\Delta$ in these experiments. It can be verified that $\Delta$ is significantly lower when the two terms in the objective function are properly balanced.

**Two-phases optimization:** Having a rather accurate estimation of human pose, we performed another experiment

[3]The dataset is available through http://users.ics.forth.gr/~argyros/research.html

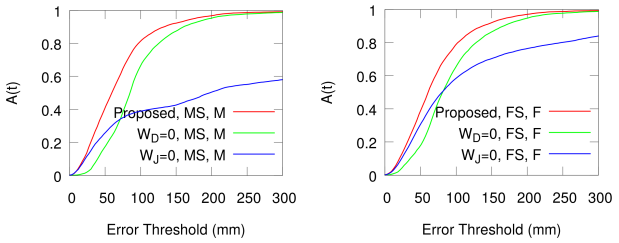



Figure 5. The accuracy $A(t)$ of the proposed method for the cases of balanced depth and joints localization terms (red curves), joints-localization-only term (green curves) and depth-only term (blue curves) for the $MS_1$ (left) and the $FS_1$ (right) sequences.

| Sequence, model | $w_D = 1.0$, $w_J = 0.0$ | $w_D = 0.0$, $w_J = 1.0$ | $w_D = 3.0$, $w_J = 1.0$ |
|---|---|---|---|
| $MS_1$, **M** | 140.2 | 98.9 | **67.3** |
| $FS_1$, **F** | 152.1 | 98.7 | **75.6** |

Table 2. Error $\Delta$ (in mm) for experiments isolating different terms of the objective function. Rows correspond to the sequences $MS_1$, $FS_1$, each of which is tracked with the proper model ($\mathbf{M_1}$, $\mathbf{F_1}$, respectively). Columns correspond to different weighting schemes (depth only, joints only, balanced). Boldface indicates best results.

| Method \ $w_D$ | 0.5 | 1.5 | 3.0 | 6.0 | 18.0 |
|---|---|---|---|---|---|
| Baseline PSO | 82.8 | 67.5 | 84.1 | 93.7 | 108.3 |
| Perturbed PSO | 77.0 | 70.0 | 67.3 | 73.5 | 81.2 |

Table 3. Error $\Delta$ for the canonical, baseline PSO versus perturbed PSO for various values of the weighting factor $w_D$.

to check whether, starting from this solution, we can further reduce $\Delta$ by employing a second, refinement phase that optimizes an objective function that consists only of the depth term. This yielded a reduction of $\Delta$ of less than 0.5 cm. This is in strong support of the objective function of Eq.(2).

**Baseline vs perturbed PSO:** As described in Section 2.4, we employ a variant of the PSO in which particles are perturbed during the optimization procedure in order to help them escape local minima. In the experimental setting of Section 3.3, we investigated the difference in performance of this perturbed version of PSO relative to the baseline, canonical version [6]. Table 3 summarizes the obtained results for different $w_D$s. For all tested values, the perturbed PSO provides more accurate pose estimates.

**The impact of the personalized human model:** We performed experiments to showcase the impact of using a personalized human body model. In that direction, we measured the error $\Delta$ when different models are used for tracking. We considered all possible combinations of sequences ($MS_1$, $MS_2$, $FS_1$, $FS_2$) with models ($\mathbf{M_1}$, $\mathbf{M_2}$, $\mathbf{F_1}$, $\mathbf{F_2}$, $\mathbf{P}$). It should be stressed that all models have been adjusted

| Sequence \ model | $\mathbf{M_1}$ | $\mathbf{M_2}$ | $\mathbf{F_1}$ | $\mathbf{F_2}$ | $\mathbf{P}$ |
|---|---|---|---|---|---|
| $MS_1$ | **67.3** | 103.7 | 85.4 | 87.5 | 95.1 |
| $MS_2$ | 83.3 | **76.9** | 84.3 | 94.7 | 112.2 |
| $FS_1$ | 90.2 | 96.9 | **75.6** | 99.4 | 100.2 |
| $FS_2$ | 84.7 | 103.2 | 92.8 | **79.7** | 108.1 |

Table 4. The error $\Delta$ when the sequences $MS_i$, $FS_i$ are tracked with models $\mathbf{M_i}$, $\mathbf{F_i}$, $\mathbf{P}$. Boldface font indicates best results.

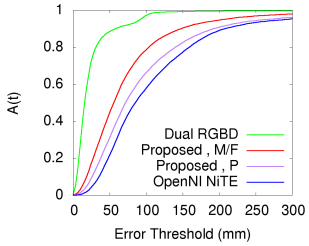



Figure 6. The accuracy $A(t)$ of the Dual RGBD method [22] (green), the OpenNI method [28] (blue) and of the proposed one when personalized (red) or primitive-based models (purple) are used (results aggregated over all four sequences of the dataset).

to fit the dimensions of the actual person. Thus, differences in performance should be attributed to the individual human body shape differences and not to their absolute scale difference. Table 4 shows the error $\Delta$ in these experiments. The minimal values appearing on the diagonal shows that a sequence is tracked more accurately when the proper (personalized) model is used. Moreover, the personalized model always outperforms the on based on primitives (**P**).

**Proposed vs [22] vs [28]:** We compare the performance of the proposed method with two existing methods. The first one [22] is referred to as "Dual RGBD" because it employs synchronized input from two, extrinsically calibrated, wide baseline RGBD cameras. This was possible because of the synthetic nature of the developed datasets that permit the rendering of a scene from different views. The second is the widely deployed OpenNI NiTE method [28]. Figure 6 summarizes the accuracy $A(t)$ of the Dual RGBD (green), the OpenNI (blue), the proposed with personalized models (red) and the proposed with a primitives-based model (purple) methods, aggregated over all four sequences. Due to the wealth of the used information, the Dual RGBD method achieves higher accuracy than any of the single RGBD camera methods. However, it requires more complex hardware setup (two synchronized and extrinsically calibrated cameras), is computationally more intensive and requires initialization. Figure 6 also shows that from the two methods that use a single RGBD sensor, the one proposed in this paper outperforms clearly the OpenNI method regardless of the model used. Still, the personalized model improves pose estimation accuracy considerably.

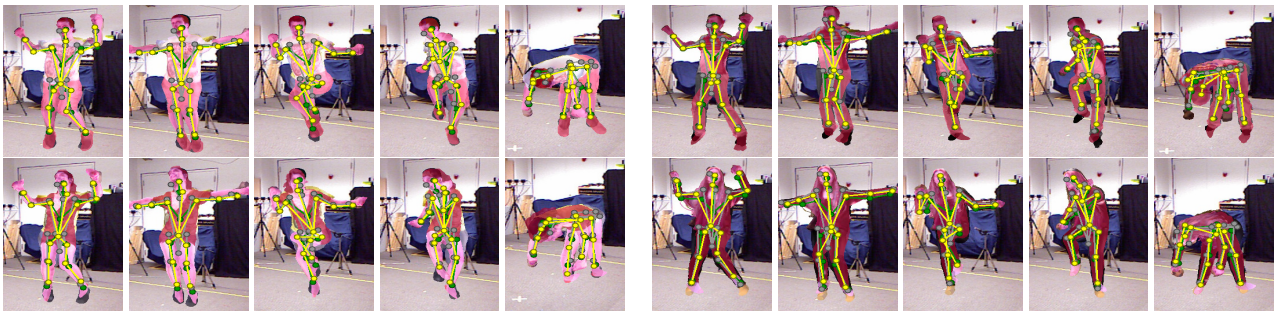


Figure 7. Sample qualitative tracking results on the synthetic sequences $MS_1$ (top-left), $MS_2$ (top-right), $FS_1$ (bottom-left), $FS_2$ (bottom-right). Grey skeletons: OpenPose proposals, yellow skeletons: proposed method, green skeletons: ground truth.

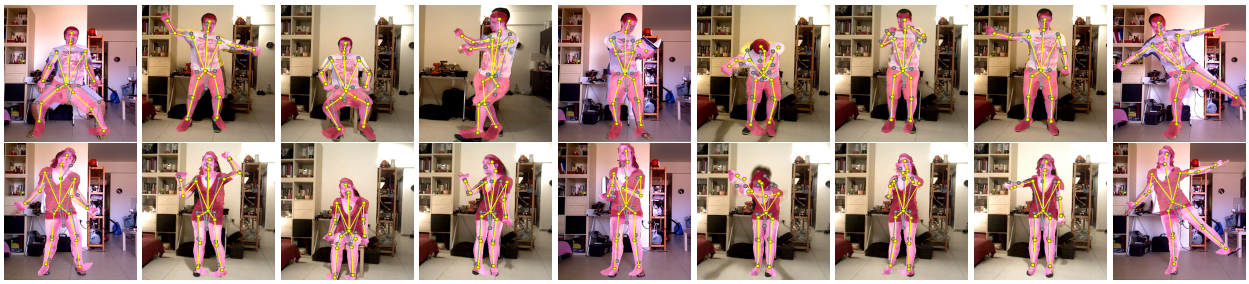


Figure 8. Sample qualitative tracking results on real sequences. Grey skeletons: OpenPose proposals, yellow skeletons: proposed method.

### 3.4. Qualitative results

Figure 7 shows results on the sequences $MS_i$ and $FS_i$ and Figure 8 results on real data[4]. The estimated skinned models are rendered on the RGB frames. It can be verified that there is a good fit between the estimated body models and the observed human figures. All experiments were performed on an Intel i7-4790 16GB RAM, NVIDIA Geforce GTX 970 GPGPU. The Open Pose 2D body joints estimation runs as a separate thread at 10 fps. The proposed method is very well suited for GPU parallelization since all particles in a PSO generation can be computed in parallel. The body models are only uploaded once and we query all particles per generation in one step. Skinning is handled via shaders and the resulting rendering output is a texture that gets directly compared to the observation using CUDA giving us scores without a slow copy to system memory. Indicatively, a simpler serial GPU rendering and CPU scoring implementation performed at 0.19 fps. The current GPU implementation operates at 9 fps.

## 4. Discussion

We proposed a new method for 3D human pose estimation based on markerless observations provided by an RGBD camera. The proposed method follows a hybrid approach that integrates *tightly* a discriminative and a generative component. The method optimizes an objective function that consists of two terms, one that registers the 3D structure of hypotheses and observations and a second that registers joint locations that are proposed by OpenPose. The combination of these two terms performs remarkably better that any of the terms in isolation. The use of the joints localization term enables automatic initialization and prevents drift. At the same time, the impact of errors in the localization of joints is minimized because of the contribution of the depth term. We also show that personalized skinned body models that can be easily and automatically acquired with off-the-shelve components, can be incorporated to the body pose estimation pipeline, resulting in increased accuracy compared to (a) adapted, generic models (either primitives-based or skinned) and (b) to competitive methods. These conclusions are supported by several experiments on a synthetic dataset which is based on the MHAD and the CMU public datasets and includes motion capture data and detailed skinned human models for a variety of human motions. Future work will investigate objective function terms that exploit color information and to extend the framework to track humans interacting with objects.

[4]Detailed results at https://youtu.be/SCgpIIaRIuI


## Acknowledgements

This work was partially supported by the EU H2020 ICT projects Co4Robots and ACANTO.

## References


[1] A. Baak, M. Muller, G. Bharaj, H.-P. Seidel, and C. Theobalt. A data-driven approach for real-time full body pose reconstruction from a depth camera. In *IEEE ICCV*, pages 1092–1099. IEEE, 2011.

[2] A. Bisacco, Y. Ming-Hsuan, and S. Soatto. Fast human pose estimation using appearance and motion via multi-dimensional boosting regression. In *IEEE CVPR*, 2007.

[3] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. *arXiv preprint arXiv:1611.08050*, 2016.

[4] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *IEEE CVPR*, 2017.

[5] L. Chen, H. Wei, and J. Ferryman. A survey of human motion analysis using depth imagery. *Pattern Recognition Letters*, 34(15):1995 – 2006, 2013.

[6] M. Clerc and J. Kennedy. The particle swarm-explosion, stability, and convergence in a multidimensional complex space. *Evolutionary Computation, IEEE Transactions on*, 6(1):58–73, 2002.

[7] S. Corazza, L. Mundermann, E. Gambaretto, G. Ferrigno, and T. Andriacchi. Markerless motion capture through visual hull, articulated icp and subject specific model generation. *IJCV*, 87(1-2):156–169, 2010.

[8] J. Deutscher and I. Reid. Articulated body motion capture by stochastic search. *IJCV*, 61(2):185–205, 2005.

[9] P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. *IJCV*, 61(1):55–79, Jan. 2005.

[10] A. Feng, D. Casas, and A. Shapiro. Avatar reshaping and automatic rigging using a deformable model, 2015.

[11] M. A. Fischler and R. A. Elschlager. The representation and matching of pictorial structures. *IEEE Trans. Comput.*, 22(1):67–92, Jan. 1973.

[12] J. Gall, B. Rosenhahn, T. Brox, and H.-P. Seidel. Optimization and filtering for human motion capture. *IJCV*, 87(1-2):75–92, 2010.

[13] J. Gall, C. Stoll, E. de Aguiar, C. Theobalt, B. Rosenhahn, and H. P. Seidel. Motion capture using joint skeleton tracking and surface estimation. In *IEEE CVPR*, pages 1746–1753, 2009.

[14] V. Ganapathi, C. Plagemann, S. Thrun, and D. Koller. Real time motion capture using a single time-of-flight camera. In *IEEE CVPR*, 2010.

[15] H. Hamer, K. Schindler, E. Koller-Meier, and L. Van Gool. Tracking a hand manipulating an object. In *IEEE ICCV*, 2009.

[16] T. Helten, M. Müller, H.-P. Seidel, and C. Theobalt. Real-time body tracking with one depth camera and inertial sensors. In *IEEE ICCV*, pages 1105–1112. IEEE Computer Society, 2013.

[17] L. Kavan, S. Collins, J. Žára, and C. O'Sullivan. Skinning with dual quaternions. In *Proceedings of the 2007 Symposium on Interactive 3D Graphics and Games*, I3D '07, pages 39–46, New York, NY, USA, 2007. ACM.

[18] N. Kyriazis and A. A. Argyros. Scalable 3d tracking of multiple interacting objects. In *IEEE CVPR*, pages 3430–3437, Columbus, Ohio, USA, June 2014. IEEE.

[19] S. Li, W. Zhang, and A. Chan. Maximum-margin structured learning with deep networks for 3d human pose estimation. In *IEEE ICCV*, pages 2848–2856, 2015.

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

[21] D. Michel and A. Argyros. Apparatuses, methods and systems for recovering a 3-dimensional skeletal model of the human body, March 2016.

[22] D. Michel, C. Panagiotakis, and A. Argyros. Tracking the articulated motion of the human body with two rgbd cameras. *Machine Vision Applications*, 26(1):41–54, 2015.

[23] D. Michel, A. Qammaz, and A. A. Argyros. Markerless 3d human pose estimation and tracking based on rgbd cameras: an experimental evaluation. In *International Conference on Pervasive Technologies Related to Assistive Environments (PETRA 2017)*, pages 115–122, Rhodes, Greece, June 2017. ACM.

[24] I. Mikic, M. Trivedi, E. Hunter, and P. Cosman. Human body model acquisition and tracking using voxel data. *IJCV*, 53(3):199–223, 2003.

[25] T. Moeslund, A. Hilton, and V. Kruger. A Survey of Advances in Vision-based Human Motion Capture and Analysis. *CVIU*, 104:90–126, 2006.

[26] I. Oikonomidis, N. Kyriazis, and A. Argyros. Efficient Model-based 3D Tracking of Hand Articulations using Kinect. In *BMVC*, Dundee, UK, 2011.

[27] I. Oikonomidis, N. Kyriazis, and A. A. Argyros. Tracking the articulated motion of two strongly interacting hands. In *IEEE CVPR*, pages 1862–1869. IEEE, June 2012.

[28] OpenNI. *OpenNI User Guide*. OpenNI organization, November 2010.

[29] P. Padeleris, X. Zabulis, and A. A. Argyros. Head pose estimation on depth data based on particle swarm optimization. In *IEEE Computer Vision and Pattern Recognition Workshops (CVPRW 2012)*, pages 42–49. IEEE, June 2012.

[30] C. Plagemann, V. Ganapathi, D. Koller, and T. Sebastian. Real-time identification and localization of body parts from depth images. In *ICRA*, 2010.

[31] G. Pons-Moll, L. Leal-Taixe, T. Truong, and B. Rosenhahn. Efficient and robust shape matching for model based human motion capture. In R. Mester and M. Felsberg, editors, *Pattern Recognition*, volume 6835 of *LNCS*, pages 416–425. Springer, 2011.

[32] R. Poppe. Vision-based human motion analysis: An overview. *CVIU*, 108(1-2):4 – 18, 2007. Special Issue on Vision for Human-Computer Interaction.

[33] D. Ramanan. Learning to parse images of articulated bodies, 2006.

[34] G. Rogez and C. Schmid. Mocap-guided data augmentation for 3d pose estimation in the wild. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, pages 3108–3116. Curran Associates, Inc., 2016.

[35] G. Rogez, P. Weinzaepfel, and C. Schmid. Lcr-net: Localization-classification-regression for human pose, 07 2017.

[36] A. Shapiro, A. Feng, R. Wang, H. Li, M. Bolas, G. Medioni, and E. Suma. Rapid avatar capture and simulation using commodity depth sensors. *Computer Animation and Virtual Worlds*, 25(3-4):201211, 2014.

[37] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-Time Human Pose Recognition in Parts from Single Depth Images. In *IEEE CVPR*, 2011.

[38] L. Sigal, M. Isard, H. Haussecker, and M. Black. Loose-limbed people: Estimating 3d human pose and motion using non-parametric belief propagation. *IJCV*, 98(1):15–48, 2012.

[39] C. Sminchisescu, A. Kanaujia, Z. Li, and D. Metaxas. Discriminative density propagation for 3d human motion estimation. In *IEEE CVPR*, volume 1, pages 390–397, 2005.

[40] S. Stefanou and A. A. Argyros. Efficient scale and rotation invariant object detection based on hogs and evolutionary optimization techniques. In *Advances in Visual Computing (ISVC 2012)*, pages 220–229. Springer, July 2012.

[41] A. Toshev and C. Szegedy. Deeppose: Human pose estimation via deep neural networks. In *IEEE CVPR*, pages 1653–1660. IEEE Computer Society, 2014.

[42] C. M. University. Cmu graphics lab motion capture database. http://mocap.cs.cmu.edu/. Accessed: 2017-06-01.

[43] R. Vidal, R. Bajcsy, F. Ofli, R. Chaudhry, and G. Kurillo. Berkeley mhad: A comprehensive multimodal human action database. In *Proceedings of the 2013 IEEE Workshop on Applications of Computer Vision (WACV)*, WACV '13, pages 53–60. IEEE Computer Society, 2013.

[44] J. Vijay, E. Trucco, and S. Ivekovic. Markerless human articulated tracking using hierarchical particle swarm optimisation. *Image and Vision Computing*, 28(11):1530–1547, 2010.

[45] T. von Marcard, G. Pons-Moll, and B. Rosenhahn. Human pose estimation from video and imus. *Transactions on Pattern Analysis and Machine Intelligence*, 38(8):1533–1547, Jan. 2016.

[46] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. Convolutional pose machines. In *CVPR*, 2016.

[47] X. Wei, P. Zhang, and J. Chai. Accurate realtime full-body motion capture using a single depth camera. *ACM Trans. Graph.*, 31(6):188:1–188:12, Nov. 2012.

[48] H. Yasin, U. Iqbal, B. Kruger, A. Weber, and J. Gall. A dual-source approach for 3d pose estimation from a single image. In *IEEE CVPR*, 2016.

[49] M. Ye, W. Xianwang, R. Yang, R. Liu, and M. Pollefeys. Accurate 3d pose estimation from a single depth image. In *IEEE ICCV*, pages 731–738, 2011.

[50] L. Zhang, J. Sturm, D. Cremers, and D. Lee. Real-time human motion tracking using multiple depth cameras. In *IROS*, Oct. 2012.