# Human Pose Estimation

Matteo Anedda, Christoph Schröter, Masoud Taghikhah

***Abstract*—As an abstraction of human body estimation, human pose estimation is a demanded but challenging research topic in computer vision. The applications range from simple use in mobile devices or entertainment systems up to medical purposes. Therefore, an estimation approach must provide accurate results to be considered valuable for its area of application. For this reason, this overview of human pose estimation first explains datasets relevant to both benchmarking and supervision approaches, followed by computational rules for measuring the quality of an estimate. This field of research is not only very broad, but can also be viewed from many different perspectives, which is why a general distinction is made between the dimensions of the estimates. To provide essential knowledge for the approaches of later sections, the parametrized human body model *smpl* is introduced and its principal concepts are subsequently documented. In 2d pose estimation, top-down and bottom-up frameworks were explained where deep neural networks aid in determining the pose. Important papers in the topic of 3d human pose estimation are presented and their architectures and results explained. Finally the current state of the broad topic human pose estimation is summarized with an outlook into possible future research.**

## I. Introduction

Human body estimation is a broad and increasingly relevant area of research that allows human movements to be detected or even predicted, and also their meshes to be created based on visual input. Due to the widespread distribution and availability of relatively inexpensive but powerful video recording hardware and computers, the researched technique of pose estimation can be used by many people in various fields of application. One such application may be in control-based domains such as virtual and augmented reality, where immersion is achieved by capturing human movements and executing them through the avatar. This can be applied not only to today's video games, but also to online business meetings, sprints or educational courses that can take place in a virtual environment. Moreover, the approaches are existential for the futuristic meta-project in which social life is also shifting into a virtual world and new forms of communication are being introduced. An further interesting aspect is also the use in autonomous machines. Here autonomous cars need to make decisions considering the near future, therefore they need to be able to predict human motion, which is tightly coupled with their pose. The same applies to robotic systems working around or even with humans. Since human posing is an indicator for their emotions aswell, social or collaborative robots could use this information in the future. Even medical information can be obtained from a person's posture and used in diagnostics and therapy or for the correct posture when sitting or doing sports. These examples show that human pose estimation is an important research area today and will become even more so in the future. However, there are challenges to overcome in order to deliver accurate estimations. Firstly, a vulnerable source of error is associated with the acquisition of the image data, which is divided into the following categories. *passive* or *active* sensing.

Passive sensing uses visible light or other electromagnetic wavelengths to acquire position data, while active sensing requires special devices attached to the person's body to obtain parametric 3d data [68]. In the initial phase of active sensing, the object whose movement is to be detected must be wired, which makes its movement cumbersome [67]. Newer systems offer more flexibility, but are prone to heading errors that accumulate over the recording period and lead to inaccuracies [110]. With passive scanning, motion detection is difficult due to the 3d-to-2d projection and the large amount of information in the image sequences [67].

Errors can occur due to different lighting conditions, backgrounds or clothing. Multiple or incomplete humans could also hamper the algorithm. Furthermore, 3d pose estimation needs to add an extra dimension of depth, which could lead to ambiguities. As neural networks, which are used a lot for pose estimation today, get bigger and used more frequently, enough and sufficient training and testing data needs to be available in the future. These datasets also need to be representative for humans in their normal environment. Annotation, especially for 3d data, can be a very resource-intensive and time-consuming task too.

## II. Benchmark Datasets

Huge data sources are part of the benchmarking process for human posture estimation approaches. The datasets used specifically for this purpose contain images showing one or more individuals in different poses, as well as other information about joint and limb positions. This information is acquired by motion capture markers on the body or using *IMU* units as passive sensing devices, attached to the body to determine the position regardless of obstacles. Existing video footage can also be manually annotated. Some datasets focus on different features within its content to ensure the quality of a model in relation to that aspect. Commonly used datasets are explained in more detail in this section.

### A. Max Planck Institute datasets

*Perceiving Systems*[1] is a department of the *Max Planck Institute for Intelligent Systems*[2] that is specialized in computer vision and, in addition to scientific publications, also

[1]https://ps.is.mpg.de/

[2]https://is.mpg.de/

provides datasets[3] for e.g. pose estimation approaches. The listed datasets can be subdivided according to the following aspects:

*1) Clothing:* The individuality of a person is expressed by his clothing. To account for this feature, a model called *CAPE* is built from 4d posture sequences of 8 men and 3 women. This dataset consists of about 80000 frames. In *CAPE*, clothing is represented by additional offsets that depend on the pose $\theta$, the clothing type $c$ and a shape variation $z$ that shift the vertices of a human body model.To regress the parameters, displacement vertices are calculated from clothed minus unclothed body scans, which are passed to a neural encoder-decoder network. [61] Further work by Ma et al. has generated a synthetic dataset on specific *CAPE* subjects and published it as *ReSynth* for researchers [62].

*2) Full-body:* Acquiring 3d scans and data from multiple people in an outdoor environment is challenging because the markers are difficult to track. Von Marcard et al. have shown in their publication that capturing sufficient data in a scene is possible with 6-17 *IMU* units attached to each person, combined with a single hand-held camera. The recorded 51000 images are available for research. [110] A similar approach is followed by Huang et al. with 17 *IMU* units for 10 subjects in 64 sequences, resulting in 330000 time instances [38]. Human-environment interaction is mainly covered in the datasets of Hassan et al. which consist of three parts in different scenes [33]. The *GRAB* dataset, on the other hand, targets the relationship between full-body models and object manipulation. It contains motion data of 10 individuals interacting with 51 objects in 4 different contexts, e.g., lifting, transferring, hand-to-hand transfer, and using objects [99]. In contrast to *GRAB*, the dataset from Müller et al. includes all of a person's interactions with themselves [70].

*3) Hand scans:* The hand contributes to communication, e.g., the hand gesture is used to confirm a statement in conversation. To incorporate this expressiveness into existing full-body models, Romero et al. developed the *MANO* model from approximately 1000 3d scans of 31 subjects in 51 poses. These scans showed female and male hands, both left and right, interacting with primitives. [88] Hasson et al. also published a synthetically generated hand dataset *ObMan* that focuses on the manipulation of grasped primitives [34].

*4) synthetic data:* A much more cost-effective approach is to create realistic body data from existing motion capture sources. An prominent example is *SURREAL* by Varol et al. which consists of 6 million frames [108]. In [37], Hoffmann et al. investigate the impact of synthetic and synthetically augmented images on the capability of neural networks to generalize to real world data. They subsequently release their datasets,
$mathbbD_M$ and $\mathbb{D}_{Style}$.

*5) Generalization of datasets:* Many different 3d scans are based on markers and motion capture software. Unfortunately, the number of markers varies from dataset to dataset, so their use as a data source for a body model leads to inaccuracies and further adjustments. A common solution to this problem is provided by the *MoSh++* algorithm and its resulting *AMASS* dataset, which unifies existing datasets in a common framework. It consists of 11265 motions from 344 subjects with 40 hours of content. [63]

[3]https://ps.is.mpg.de/research_fields/datasets-and-code

### B. COCO dataset

According to Lin et al. the context of a given scene has an impact on the quality of the estimation. Therefore, images with many classifiable objects, such as animals, people, etc., are the content of the dataset to encode contextual information about their constellation and appearance in images. In addition, mainly non-iconic images of no centered objects are included. The dimension of *COCO* is 328000 images, divided into 91 object categories with a total of 2500000 label instances. [55]

### C. Caesar dataset

[86]

## III. Benchmark metrics

The quality of the estimated poses and thus of the applied algorithm is assessed by metrics that define how to measure the deviation of the prediction from ground truth. The common approach is to first extract the features from an estimated pose, which are then compared with the features calculated from ground truth. The measured values can be used to assess how precise the pose estimate is. For this analysis, the previously mentioned datasets from section II are used as reference. Some metrics have their own thresholds that define whether an estimate is correct or deviates too much from reality and is therefore false.

### A. Percentage of Correct Parts PCP

This criterion encompasses a comparison of the recognized and recognizable body parts. The definition of a correctly recognized limb includes both the distances $l_1, l_2$ of its endpoints from those contained in the dataset and its total length $L$. Figure 1 shows an example with the left (green) line indicating information from the dataset that is compared to the estimated right line. Another factor $p$ multiplied by $L$ defines the threshold value to which $l_1$ and $l_2$ are compared. If $l_1$ or $l_2$ exceed the threshold, the body part is not detected correctly, resulting in a lower $PCP$ score. The smaller p, the stricter the evaluation and thus higher the accuracy. [22]

### B. Percentage of Detected Joints PDJ

The metric *PCJ* addressed in this section is similar to the already established *PCP* methode. While *PCP* depends on individual limb lengths, *PCJ* uses the torso length $T$ as a global reference, so that a body part is correctly detected if both $l_1$ and $l_2$ do not surpass the threshold given by $T$ and a factor $p$. [106]

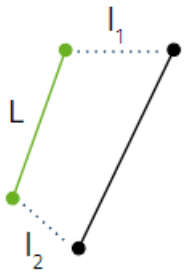


Fig. 1. Abstract visualisation of the values required for the PCP calculation. A solid line between two endpoints represents a bone in the ground truth of a dataset in green and the detected bone in black. The dashed lines show the distances $l_1, l_2$, which are evaluated with the true length $L$ of the bone.

### C. Percentage of Correct Key-points PCK

For this particular criteria, the maximum bounding box length $B$ must be calculated from the existing dataset information. The *PCK* score is calculated analogously to the PCP and PDJ metrics, where the threshold is defined by $B \cdot p$. [115]

### D. Head-normalized Probability of Correct Key-points PCKh

In *PCKh*, the following change to the standard *PCK* was proposed by setting the threshold to $H \cdot 0.5$, which corresponds to half of the head frame. Since the maximum length of the bounding box depends on the position of the individual body parts, the head box is chosen so that the threshold is not influenced by the subjects' articulation. [1]

### E. Object Key-point Similarity OKS

OKS measures the overlap between the extracted keypoints of the pose estimate and the grount truth and thus whether the predicted keypoints are close to reality. Equation 1 illustrates the simplified OKS score[4], which is a sum of $n \in \mathbb{N}$ detected and ground truth key-points. The parameters $d \in \mathbb{R}$ are the Euclidean distance between the corresponding ground truth value and the detected key-point, while $s \in \mathbb{R}$ denotes the object segment area and $k \in \mathbb{R}$ is a constant describing a falloff.

$$OKS = \sum_{i=1}^{n} e^{-\frac{d_i^2}{2 \times s^2 \times k_i^2}} \tag{1}$$

Optimal predictions have a high OKS value, while low values indicate poor predictions.

### F. Mean Per Joint Position Error (MPJPE)

Equation 2 shows the formula for one of the most import criterias in terms of evaluating results from 3d human pose estimation: the Mean Per Joint Position Error or short MPJPE [118]. As the name suggests, it calculates the mean of the difference between actual and estimated joint positions. $N$ denotes the number of estimated joint positions $J_i^*$ and ground truth joint positions $J_i$.

[4]https://cocodataset.org/#keypoints-eval

$$MPJPE = \frac{1}{N} \sum_{i=1}^{N} \| J_i - J_i^* \|_2 \tag{2}$$

There are versions of this criteria too, such as PMPJPE, where the estimated pose gets rigidly aligned to the ground truth first. Thereby the P stands for procrustes transformation, which is a commonly used alignment method using procrustes analysis as described in [30]. With NMPJPE, the Normalized MPJPE, a scaling factor is applied to the estimation to minimize the sqared distances to the truth, thus allowing the metric to be more independent of the subjects body size [85].

## IV. Models in Human Estimation

This section is dedicated to human models, which are besides being a representative concept, also central aspects on which iterative pose estimation approaches are based. While simple stick models, where the human structure is modeled by a few vertices, show the joints and limbs in 2d estimation, 3d models, originally built from simple primitives such as cylinders, have evolved into a vertex structure showing the shape, muscles and body structure of a human. To conclude, this section explained in detail the human body models commonly used in 3d estimation methods.

### A. Skinned Multi-Person Linear model SMPL

In *SMPL* [57] a model $M(\vec{\beta}, \vec{\theta}, \phi)$ is learned from the *Caesar* 3d scans explained in section II, that returns a mesh from the input. This formulation is also included in Equation 3, where $\mathbb{R}^{3N}$ is a vector of $N$ = *6890* vertices sculpturing the mesh. In this formula, $\vec{\beta}$ is a vector that contains the coefficients for the shape blend shapes $\mathbb{S}$. A single shape blend $S \in \mathbb{S}$ is a vector of vertex shifts corresponding to a particular feature of a human, e.g., height or weight. $\vec{\theta}$ is a vector of pose parameters, e.g. the concatenated joint rotation in axis-angles representation, of all joints and the global orientation, which are also used as coefficients for pose blend shapes, and $\phi$ describes other learned parameters.

$$M(\vec{\beta}, \vec{\theta}, \phi) : \mathbb{R}^{|\vec{\theta}| \times |\vec{\beta}|} \mapsto \mathbb{R}^{3N} \tag{3}$$

*SMPL* is based on vertex skinning and blend shapes. A vertex changes its position depending on the motion of the associated joint. This displacement is controlled by assigned blend weights. A vector $T \in \mathbb{R}^{3N}$ of vertex positions describes a gender neutral initial human model, while a matrix $W \in \mathbb{R}^{N \times K}$ represents the blend weights per vertices and $K$ = *23* joints. The joints that describe the human structure and form the skeleton are represented by 3d positions. Moreover, $T$ can be rearranged by the pose-blending function $B_P(\vec{\theta})$ according to the given pose parameter, leaving its initial shape unaffected, while $B_S(\vec{\beta})$ reshapes the identity model by its given shape blend coefficients. The two terms $B_P(\vec{\theta})$ and $B_S(\vec{\beta})$ thus serve as offsets to the initial mesh $T$, which are included in the sum $T_F(\vec{\beta}, \vec{\theta})$. Finally, a skinning function $W(T_F(\vec{\beta}, \vec{\theta}), J(\vec{\beta}), \vec{\theta}, \mathcal{W})$ can be created containing the initial model $T$ transformed by $T_F$, the joint positions calculated

from $J(\vec{\beta})$, the rotations $\vec{\theta}$ and the blending weights $\mathcal{W}$, shown in Equation 4. Since $\beta$ affects the shape of a person and thus the position of his joints, the joint positions must be calculated after the shape displacements $B_S$ have been applied to the initial model $T$. The vertices in *smpl* have a strict structure so that the position of a joint can be determined by the vertices in its region. Therefore, a joint regression matrix is learned to assign weights to these particular vertices that affect joint position, which were then linearly combined to define its location.

$$M(\vec{\beta},\vec{\theta}) = W(T_F(\vec{\beta},\vec{\theta}), J(\vec{\beta}), \vec{\theta}, \mathcal{W}) \mapsto \mathbb{R}^{3N} \tag{4}$$

As mentioned previously, a model is determined of its initial state $T$ and its offsets in $T_F$. In $T_F(\vec{\beta},\vec{\theta})$, $T$ is considered as a known parameter, while the pose correctives $B_P(\vec{\theta})$ and the reshapes $B_S(\vec{\beta})$ are added. The shape of a single human can be viewed as a sum of shape blend shapes $S \in \mathbb{S}$ multiplied by their corresponding coefficients in $\beta$, describing the impact of the associated features on the resulting mesh. This concept is expressed in Equation 5.

$$B_s(\vec{\beta},\mathbb{S}) = \sum_{n=1}^{|\vec{\beta}|} \beta_n \cdot S_n \tag{5}$$

With respect to Equation 6, $B_P(\vec{\theta})$ can be viewed as linear combinations of pose blend shapes $\mathbb{P}$ learned from *Caesar* dataset.

$$\begin{aligned} &T_F(\vec{\beta},\vec{\theta}) : \mathbb{R}^{|\vec{\theta}|\times|\vec{\beta}|} \mapsto \mathbb{R}^{3N} \\ &T_F(\vec{\beta},\vec{\theta}) = T + B_S(\vec{\beta}) + B_P(\vec{\theta}) \\ &\qquad = T + B_S(\vec{\beta}) + \sum_{n=1}^{9K}(R_n(\vec{\theta}) - R_n(\vec{\theta^*}))P_n; \end{aligned} \tag{6}$$

$\theta$ is a concatenation of all $K$ rotations of the joints plus the root orientation. These rotations are represented as axis-angles, which can be converted into $\mathbb{R}^{3\times3}$ rotation matrices using the *Rodrigues* formula[5]. Each element of this matrix can be stacked into a 9d vector, applying it to $K$ joint yields a 207d vector. This entire concept is described by the function $R : \mathbb{R}^{|\vec{\theta}|} \mapsto \mathbb{R}^{9K}$. In Equation 6, $R_n(\vec{\theta^*})$ denotes the nth element of the 207d vector resulting from the residual position of $T$, while $R_n(\vec{\theta})$ denotes the nth element of the entered pose rotations $\theta$. The effect of the learned pose-blend shapes thus depends on the deviation between $\theta$ and $\theta^*$ and is neglected if the nth joint orientation element corresponds to the initial orientation of the nth joint rotation element in $T$.

Equation 6 is used in skinning, in general to calculate a new vertex position based on the shape and pose parameters. A previous skinning approach is called *linear blend skinning*. *Linear blend skinning* is an algorithm that first rotates the calculated joints based on the pose parameters in $\theta$ to create the preferred pose, with each new joint calculated as a homogeneous world transformation $G_k(\vec{\theta}, J)$. Furthermore, the vertices follow the movement of the joints according to how closely a vertex is coupled to them, defined by its weight in $\mathbb{W}$, to form the skin of the human in its new pose. In Equation 7 this concept is represented as the sum of all $K = 23$ transformations $G_k(\vec{\theta}, J)$ applied to the vertex in its old position $\bar{t}_i$ multiplied by its weight $w_{k,i}$ to scale the applied transformation appropriately for the influence of the joints on the vertex. $G'_k$ is obtained by removing the rest pose transformations $G_k(\vec{\theta^*}, J)$ from the new joint transformations $G_k(\vec{\theta}, J)$.

$$\bar{t}'_i = \sum_{k=1}^{K} w_{k,i} \cdot G'_k(\vec{\theta}, J) \cdot \bar{t}_i \tag{7}$$

Unfortunately, this approach is error-prone and requires additional refinement by the artists to reduce artifacts. To overcome these problems, the skinning method in *SMPL* can be rewritten by the formulations in Equation 4 and their application in Equation 6. This Equation 8 is formulated below, considering both $b_P$ and $b_S$ as vectors based on $B_P$ and $B_S$, which offset the vertex position $\bar{t}_i$.

$$\bar{t}'_i = \sum_{k=1}^{K} w_{k,i} \cdot G'_k(\vec{\theta}, J(\vec{\beta})) \cdot (\bar{t}_i + b_{S,i}(\vec{\beta}) + b_{P,i}(\vec{\theta})) \tag{8}$$

## V. 2D POSE ESTIMATION

The foremost step in Human Pose Estimation (HPE) is to model the human body entity representing its kinematic structure and shape information. In recent studies, benefiting from the intuitive nature of graph representations, the human body structure has been characterized by anatomical joints and their positions [121]. This approach is known as the kinematic model, and is widely employed to describe human poses. In 2D Human Pose Estimation, the goal is to localize the 2D position or spatial location of human body keypoints (kinematic joints) from the input data [82].

Recently, deep neural networks have achieved a significant breakthrough in 2D HPE and are mainly employed to extract robust features for keypoint recognition and localization directly from the input data (images and videos). Since the quality of feature representation is intimately tied to network architecture, the subject of network design was chosen to be thoroughly investigated.

### *HPE Network Architecture Design Challenges*

Like many tasks in computer vision (e.g., object detection, image segmentation, etc.), the primary technological challenges in developing 2D HPE algorithms are the accuracy and efficiency of the proposed method. High precision pose estimation facilitates accurate human body information for subsequent tasks, including lifting from 2D to 3D HPE models. In addition to the accuracy of an HPE algorithm, its efficiency (inference speed) is also desired for real-time applications. However, there is mostly a trade-off between accuracy and

[5]https://mathworld.wolfram.com/RodriguesRotationFormula.html

efficiency concerning algorithm development since the high accuracy methods tend to be more complex and demand powerful resources for computation. Consequently, lightweight models with comparable precision are more desirable for mobile or wearable devices.

Another crucial challenge in 2D HPE is estimating poses for multi-person scenarios. 2D single-person HPE localizes joint coordinates when only one human instance is in the input image, but for images containing multiple persons, the HPE model needs to recognize each human in the input image before estimating its keypoints positions. This process can be done by a two-stage process called the top-down approach that first detects individual persons, i.e., the input image is divided into sub-images (patches) enclosing just one person [94]. Then the deep learning keypoint detection algorithm can be applied to each patch. On the other hand, there is an alternative group of approaches known as bottom-up, in which human joints are detected directly from the image without an explicit object detection stage [16]. The following sections will review current frameworks utilizing top-down and bottom-up approaches to their network architectures.

### A. Top-Down Framework

The top-down approaches employ human body detectors [66], [84] to obtain a set of bounding boxes (each corresponding to one person) from the input images and then perform pose estimation within each bounding box. The existing literature in 2D HPE based on top-down approaches can also be divided into fine-grained sub-categories discussed in this section: regression-based and heat map-based approaches.

*1) Regression-based methods:* There are many works based on the regression technique [**?**], [10], [23], [26], [52], [81], [95], [97], [107], [111] that directly regress human body joint coordinates via an end-to-end trained network that maps the input image to the pre-defined graphical structures (kinematic keypoints). Accordingly, the regression hypothesis is defined as follows:

Given an image I, the goal of pose estimation is to predict a possibly empty set of human instances, $\{Pi\}_{i=1}^{N}$ where $\boldsymbol{N}$ is the number of persons in the image. For each person, we need to predict its bounding box information, $\{Bi\}$, as well as its keypoint coordinates, $Ki = \{(x_j, y_j)\}_{j=1}^{J}$, where $\boldsymbol{J}$ is the number of pre-defined joints in each dataset [50].

DeepPose [107] is a regressor that explicitly predicts human joint locations through fully connected layers of a deep neural network. AlexNet [48] act as the backbone of DeepPose's cascaded architecture for feature extraction. Due to the remarkable performance of DeepPose in efficient keypoint detection by using convolutional neural networks (CNNs), human pose estimation techniques have gradually migrated from the conventional graphical models [25], [105] like Markov Random Field (MRF) [51] to deep learning approaches.

Carreira et al. [10] proposed a model based on GoogLeNet [98] that adopts an iterative feedback mechanism called Iterative Error Feedback (IEF) to enhance the efficiency of hierarchical feature extractors such as Convolutional Networks (ConvNets) in early layers. This self-correcting model progressively adjusts an initial estimate of joint coordinates by feeding back predictions error in the IEF process instead of directly predicting the coordinates in a feed-forward network.

Transformer models [109] based on the self-attention mechanism have significantly advanced the field of representation learning and promoted visual understanding tasks such as object detection frameworks that are free of region proposals, anchors, and post-processing (non-maximum suppression) procedures. In a recent study [50], authors took full advantage of the tokenized representation in transformers with self-attention layers. They propose a regression-based human pose estimation method, called pose recognition Transformer (PRTR), using two types of cascade transformers based on an end-to-end object detection Transformer (DETR) [9]. In this proposed model, the second transformer is a regressor that predicts joints' coordinates within each image patch detected from the first transformer and performs multi-person pose estimation. PRTR reveals competitive results for pose recognition compared with other existing regression-based methods on the challenging COCO dataset.

*2) Heatmap-based Methods:* In the field of human pose estimation, heatmap-based approaches seek to train a deep neural network to approximate the locations of body parts and joints based on heatmap representations [14], [72], [113]. Accordingly, during training, the pose estimation model generates J heatmaps $\{Hj\}_{j=1}^{J}$, as a 2-dimensional Gaussian distribution centered at the ground-truth joint location, where J is the total number of joints. The pixel value $Hj\{(x_i, y_i)\}$ in each joint heatmap encodes the probability that the $j^{th}$ joint lies in the location $(x, y)$ [104], [105].

The probability values provide richer supervision that facilitates the training of convolutional networks and significantly improves the performance of heatmap-based over regression-based approaches [4], [6], [29], [54], [72], [104]. Therefore, heatmaps preserve the spatial location information of joints and reduce the number of false-positive cases, which results in an increasing attraction to develop deep learning pipelines for HPE based on heatmap representations [10], [58], [107], [113].

Ramakrishna et. al [83] presented a sequential prediction framework based on convolutional networks named Convolutional Pose Machines (CPM) that learns rich implicit spatial structures by utilizing the inference machine framework to incrementally refine estimates of the human body part (e.g., head, leg, etc.) locations in multiple stages. In this method, a predictor is trained to predict the confidence of the body part locations in each stage. In the first stage, the predictors estimate the confidence of each part location from heatmaps computed on the input image patch. Since a heatmap produced only from image features is noisy and has multiple modes, in the subsequent stage, the network iteratively refines the estimated confidences using contextual information from outputs of the previous stage.

In a further expanding of Pose Machine architecture, [113] employed sequential convolutions to implicitly model long-

range spatial relationships between different parts. It is worth mentioning that multiple stages increase the network depth and make it hard to train because of the vanishing gradient problem —as the gradient is back-propagated to the layers of earlier stages, repeated multiplication makes the gradient infinitely small. Thus, as the network goes deeper, its performance gets saturated or degrades rapidly.

Before ResNet [36], there were several methods to deal with the vanishing gradient issue; for instance, the authors of this work [113] developed an auxiliary loss in a middle layer as extra supervision. But this can not tackle the problem entirely since each stage will fail to extract robust semantic features from the input data and is prone to overfitting. The core idea of ResNet is introducing "shortcut connections" that skip (jump over) one or more layers. These skip connections effectively simplify the network and reduce the impact of vanishing gradients as it allows the back-propagation of training errors at deeper levels (addressing the issue of iterative architectures with multiple stages). Residual networks dramatically improved the 2D HPE process, and numerous models [**?**], [7], [15], [18], [45], [56], [72], [92], [94], [114] have been developed due to their advantage.

An encoder-decoder network called "stacked hourglass" (SHG) based on the sequential stages of pooling and upsampling layers was presented by Newell et al. [72] to capture spatial correlations between the human body keypoints by combining low and high-resolution feature representations. Several more complicated variants have been introduced following the initial success of SHG architecture; specifically, Chu et al. [18] developed novel Hourglass Residual Units (HRUs) extended by a side branch of filters with larger receptive fields that learn features across various scales. Yang et al. [114] replaced the residual units in SHG with multi-branch Pyramid Residual Modules (PRMs) to enhance deep convolutional neural networks by constructing scale invariance features.

Cai et al. [7] recently proposed the Residual Steps Network (RSN), which efficiently maintains rich low-level spatial information by aggregating intra-level features with the same spatial size to localize the keypoints precisely using delicate local representations. In addition, they devised a novel attention mechanism, Pose Refine Machine (PRM), that refines the keypoint locations by finding a trade-off between the contribution of local and global representations in the resultant feature. Their approach accomplished state-of-the-art results on COCO and MPII benchmarks and won 1st place in the COCO Keypoint Challenge 2019.

*Top-Down Approaches Summary*

The primary components of top-down frameworks in human pose estimation consist of an object detector and a pose estimator to predict the joint positions of the human body. The object detector determines human proposal detection performance and influences pose estimation. On the other hand, the pose estimator component is the framework's heart that directly affects the pose estimation accuracy. The main reason for adopting heat map-based and regression-based methods for the pose estimator component is the speed-accuracy trade-off.

In regression-based human pose estimation, the problem is formulated as a regression one in which features extracted from convolution layers of a CNN are regressed to directly predict joint coordinates of the body parts. The regression-based networks can be trainable end-to-end and are efficient in real-time applications since they have fewer intermediate non-differentiable steps [21], [24], [28], [56]; however, they typically perform less accurately than heat map-based approaches. Because when the body parts are not completely visible, they fail to accurately estimate the locations of human body joints. To address this shortcoming, probabilistic heatmaps are employed to learn a complex mapping from occluded part appearances to joint coordinates instead of directly regressing them.

Human pose estimation based on heatmaps comprises pre-processing and post-processing procedures to encode keypoints' ground truth (GT) into heatmaps and decode heatmaps to predict joint locations. Consequently, they can be optimized easier and in comparison to regression-based methods, have a more robust generalization that delivers substantial performance, primarily suitable to be adopted when accuracy is the priority [79]. Nevertheless, heatmap-based frameworks have various heuristic network designs that are mostly not end-to-end learnable and suffer from several shortcomings: Firstly, to generate the 2D heatmaps, computationally expensive upsampling operations (e.g., deconvolution layers in [117]) are required. Also, an extra post-processing step for reducing projection errors from heatmap to ground truth coordinates is unavoidable in further keypoint estimation refinements. This makes them sub-optimal, i.e., high resolution images improve the acuracy of heatmap-based methods while increasing their computing cost.

### *B. Bottom-Up Framework*

After analyzing the top-down frameworks in the network architecture design context for human pose estimation, it is noteworthy to consider the bottom-up approaches. These approaches (e.g., [16], [26], [39], [40], [43], [47], [66], [71], [75], [80], [102] aim to perform two main tasks: firstly, human keypoint detection through extracting local features from the input image and proposing a set of human body joint candidates, and then grouping those candidates to build pose representations for each individual person.

The main distinguishing characteristic of the bottom-up approach from the former approach is that the framework design does not rely on a human detection component to predict human bounding boxes separately. As a result, the computational overhead decreases substantially by directly estimating human poses from extracted features; however, the major challenge is an identification mechanism for estimated keypoints to associate different subsets of candidates to each individual human body. Accordingly, we can divide research studies on the Bottom-Up HPE into three subgroups as human center regression [27], [75], [76], part field [40], [43], [47],

[49], [64], and associate embedding [16], [42], [59], [71] approaches.

*1) Human Center Regression:* In the human center regression-based approach, a center point is defined as representative of humans. In a study, the authors developed a novel model called Pose Partition Network (PPN) [74] with the stacked hourglass architecture [72] as the backbone that simultaneously detects keypoints and performs dense regressions from global joint candidates to robustly generate joint partitions within a specific embedding space parameterized by centroids of persons. PPN improved multi-person pose estimation using a local greedy inference approach, resulting in low computation cost and accurate joint detection.

One year later, Nie et al. presented the advantage of a Single-stage Pose Machine (SPM) [75] using the Hourglass network architecture in enhancing the efficiency of multi-person pose estimation over the conventional two-stage approaches. To achieve this, they proposed a novel Structured Pose Representation (SPR) model in which the root (centered) keypoints indicate each human body instance, and then other body joint locations are encoded into their displacements w.r.t. the roots. Accordingly, SPM demonstrated state-of-the-art efficiency and outstanding accuracy for multi-person 2D and 3D human pose estimation compared to other methods on multiple benchmark datasets such as COCO.

Recently, Geng et al. [27] studied the keypoint regions for regressing keypoint positions accurately. They employed a multi-branch network that indicates the human body instances by predicting a human center map and densely estimates a candidate pose at each pixel q within the consequent map. This direct regression approach is called Disentangled Keypoint Regression (DEKR) and shows higher accuracy than the superior bottom-up pose estimation methods on the COCO dataset.

*2) Part Field:* The part-field method is another subgroup of the bottom-up approaches pioneered by a model named OpenPose [8], which first detects keypoints and connections between them and then performs keypoint grouping according to the connective intensity between different keypoints. This approach uses Convolutional Pose Machines [113] and proposes a two-branch multi-stage architecture, where the detector branch predicts keypoints' coordinates via heatmaps. In parallel, the other represents the Part Affinity Fields (a set of 2D vector fields with vector maps) to denote the position and orientation of human body parts and associate them with each person. As a result, OpenPose achieved a real-time performance among bottom-up multi-person human pose estimation methods regardless of the number of persons in the image; however, it showed an inferior performance with low-resolution images and occlusions.

To address this problem, many studies were proposed to improve the OpenPose structure. For example, Zhu et al. [120] adopted redundant edges to increase the intensity of connections between joints in PAFs and obtained better performance than the baseline approach. Kreiss et al. [47] proposed a method called PifPaf that employed a Part Intensity Field (PIF) to localize body parts, along with Part Association Field (PAF) to associate body parts with each other. This method outperformed previous OpenPose-based approaches based on a box-free, single-shot, fully convolutional network architecture.

*3) Associate Embedding:* Motivated by OpenPose [8] and stacked hourglass structure [72], the associate embedding approach in bottom-up human pose estimation was initially introduced by Newell et al. [71], where an embedding vector is assigned to each detected keypoints and acts as an identifier tag for associating keypoints to its human body instance. They trained a single-stage deep convolutional neural network that performs keypoint detection and grouping simultaneously. Their results exhibited a remarkable performance on the COCO dataset. Inspired by the keypoint grouping procedure in OpenPose, Cheng et al. [16] devoted to improving the bottom-up pose estimation for small human body instances in the input image by suggesting an extension of HRNet, to yield high-resolution features pyramids. Their approach, Higher Resolution Network, deconvolves the high-resolution heatmaps generated by HRNet and tackles the scale variation problem in bottom-up multi-person HPE.

Jin et al. [42] proposed a multi-person pose estimation and tracking pipeline that encloses two sub-networks, SpatialNet and TemporalNet. In this approach, body part detection, keypoint embedding prediction, and part-level data association are driven by the SpatialNet, while the TemporalNet is responsible for pose tracking. One year later, the authors proposed a novel method, called Differentiable Hierarchical Graph Grouping [43], to learn the human part grouping based on alternative representations of keypoint connection for keypoint grouping. They argued that the separate keypoint grouping mechanism in bottom-up approaches is not end-end trainable, making their performance sub-optimal. To solve this issue, they modeled the human body pose using graph-structured data (nodes and edges) to represent keypoints and their relationships and deliver a fast and scalable (computationally efficient) end-to-end trainable network for detecting and grouping human body joints.

According to a recent study [59], bottom-up methods based on heatmap regression suffer from significant variance in human scales and labeling ambiguities of ground truth heatmaps because they are usually generated by 2D Gaussian kernels that have fixed standard deviations. To tackle these issues, the authors proposed the scale-adaptive heatmap regression (SAHR) method along with the weight-adaptive heatmap regression (WAHR) to deliver a model that is more resilient to various human scales and labeling ambiguities via adaptively adjusting the standard deviation for each keypoint. As a result, this framework improved the accuracy of human pose estimation by +1.5AP and outperformed the state-of-the-art top-down methods on the COCO dataset with 72.0 AP.

*Bottom-up Approaches Summary*

Top-down approaches deliver high performance in human pose estimation; however, the number of individuals in the input image directly affects their computation efficiency.

Bottom-up methods are usually faster than top-down methods since they first locate all keypoints and then group them to their corresponding person; thus, they are computationally more efficient without requiring to estimate the pose for each person separately.

### C. 2D Human Pose Estimation Summary

In summary, with the advent of deep neural networks, methods such as DeepPose [107], Stacked Hourglass Network [72], and OpenPose [8] enhanced the performance of 2D HPE significantly; however, several challenges like human body detection in crowd scenarios [15] still need to be further addressed in future research. In top-down 2D HPE methods, human body detection modules may fail to recognize different persons due to the non-max suppression mechanism that suppresses a valid instance when multiple human bodies are spatially close in the input image, and their boundaries are highly overlapped. Likewise, the keypoint grouping procedure in bottom-up approaches is prone to fail in occluded scenes. Another challenge is computing speed. Although some methods like OpenPose [8] can achieve near real-time processing on GPU processors (22 FPS), they are not still efficient for resource-constrained devices, e.g., mobile phones.

## VI. 3D POSE ESTIMATION

From a historical perspective, a 3d motion capture algorithm consists of 4 sequential processes: initialisation, tracking, pose estimation and recognition. Initialization involves both camera and model initialization, i.e. setting the camera calibration and finding a model that represents the subject and assigning its initial pose manually or automatically. Model-based approaches can be viewed iteratively, with each frame of the data source representing an iteration in which the initial pose is refined. Tracking is concerned with the relationship between the parts of the subject's body. This leads to segmentation of the subject from the background, representation changes, and establishing tracking in further images. The next phase, which is mainly covered in this section, is the estimation of the pose. A distinction is made between model-based and non-model-based methods, with the former requiring *a priori* a model. In that approaches, especially human pose estimation, a human model is used to benefit from its encoded information. This model can either be used indirectly, considering e.g. only general aspects such as size and structure, or it is a direct used model. Directly used models are both more detailed and offer broader benefits in regards to occlusion handling and embedded kinematic constraints. In an application, the observed object is approximated by the model, which is continuously refined with further images. Lastly, the captured motion is classified within the recognition, the last final phase of 3d motion capture. The concept of recognition is ether based in reconstruction or directly without any preprocessing. Further investigation [93] into direct recognition revealed erroneous estimates due to the inversion of the image data, indicating that some form of preprocessing is required to eliminate false detections. In static reconstruction, single frames are compared with pre-recorded data to detect postures, such as pointing, standing and sitting. [68]

As with 2d Pose Estimation, neural networks, particularly convolutional networks, have successfully been used to achieve more accurate results than earlier methods [3], [11], [13], [65], [103], [112], [116]. Since neural networks can be very resource intensive to compute and the architecture can be extremely complex when working with 3D data, many of the presented approaches use 2D Pose Estimation followed by an uplifting process to 3 dimensions. Also, a lot of 2D training data is available in comparison to 3D data which could be used to train a neural network, because the annotation process is way harder in higher dimensions. Especially the lack of in-the-wild 3D training data, which is not created under laboratory conditions, can be partly overcome by using 2D pose estimation methods (which work way better in-the-wild because of more training data available) first before adding an extra dimension. Therefore [13] presents a process to synthesize training images and shows that neural networks training with data generated by their method are even more effective than neural networks which were trained using real images. [87] presents a similar approach.

### A. Lifting from 2D to 3D pose

For uplifting, recent work has proven statistical models such as (deep) neural networks themselves [65], [103], matching the estimated 2d pose with a database [11] or triangulation using multiple viewpoints [20] useful. In particular [65] shows that even very simple deep neural networks can be extraordinarily effective for uplifting 2d to 3d pose estimations, considering both computational resources and failure rate.

Inspired by various 2d human pose estimation algorithms, many studies have employed the outputs of 2d pose estimate methods for 3d human pose estimation to improve in-the-wild generalization performance. For example, Martinez et al. [65] pioneered the research on lifting 2d poses to 3d space with a simple yet effective neural network. Other methods [31], [77], [101], [119] focus on fusing 2d joint heat maps from the top-down 2d pose estimation methods with 3d image cues to reduce ambiguity. The uplifting approach also makes it possible to project the calculated 3d pose to 2d again to make sure results are consistent. [112] In the following two important papers in this category are presented, the first one being one of the most impactful works of the last years from Martinez et al. as it showed how performant even simple neural networks can be. The second paper presented proposed a different, unsupervised method which uses generative adversarial networks and goes to show that other methods are still being experimented with having very good results too.

*1) A simple yet effective baseline for 3d human pose estimation:* While trying to investigate common errors in the uplifting process, [65] created a method with state-of-the-art results for 3d pose estimation using a very basic neural network with recently proposed optimization methods, whose structure is shown in Figure 2. Linear layers changing the input and output dimensions are not shown. 2d joint positions

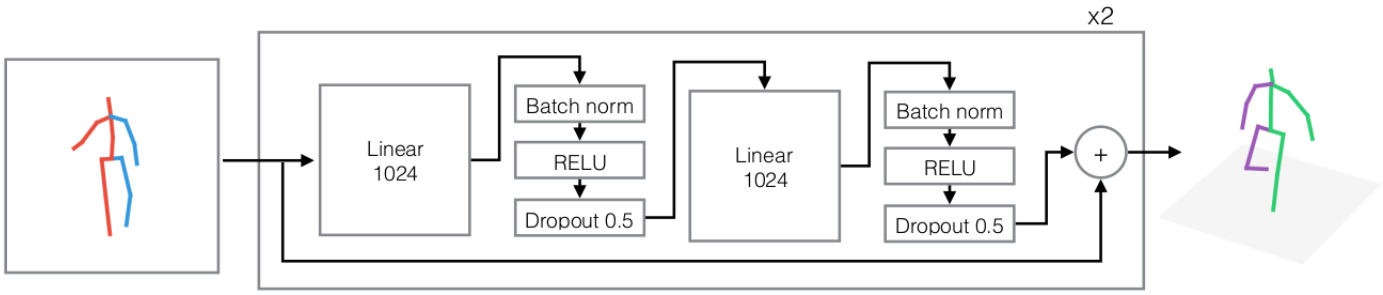


Fig. 2. Neural network structure from [65]. It consists of two inner blocks containing a linear layer followed by batch normalization and dropout. A residual connection is added from the input of the first inner block to the output of the second. This structure is then repeated another time. The networks inputs are 2d joint positions, which it outputs uplifted to three dimensions.

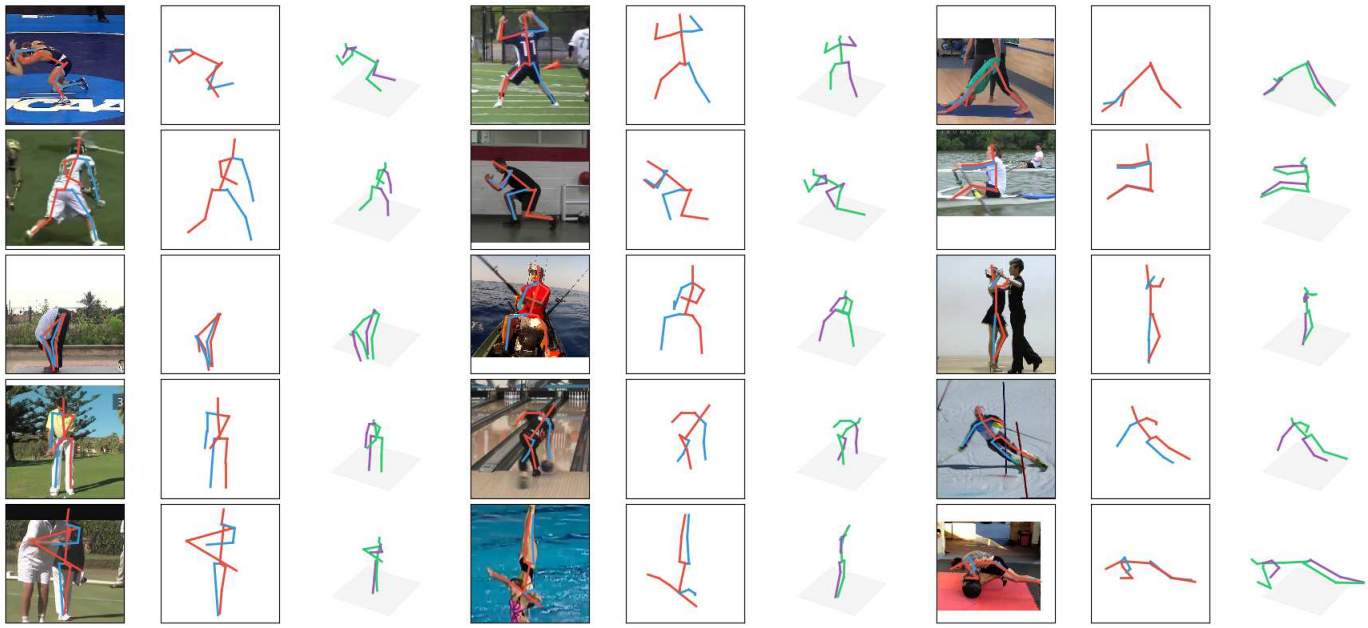

Fig. 3. Test cases from [65]. The picture on the left is the original overlayed with the 2d pose estimation (shown in the middle), while the respective uplifted 3d pose is shown on the right. Most test cases yield good results, but there are some failures too as shown in the bottom row.

are used as input data determined by the so called stacked hourglass network as described in [73], while the output consists of 3d joint positions.

Since the 2d joint positions are low-dimensional and therefore no highly complex computation is necessary, a simple linear layer with a RELU activation function is used first, followed by batch normalization and dropout as presented in [91] to prevent overfitting and improve result quality at the cost of a slight increase in computation time during training and testing. This structure is then repeated and a residual connection to the output added. These connections are proposed to help improve performance, reduce training time and lower error rates in [35]. The whole network established so far is doubled to complete the architecture which in sum consists of 4 to 5 million trainable parameters. It was then trained on Human3.6M, a dataset with 3.6 million 3d poses of humans during normal activities such as eating or walking [41].

Testing results show that this approach outperformed previous methods like [78] in most cases, despite the simple architecture that was used. However, testing on the MPII Human Pose Dataset [2] also revealed limitations of the network shown especially in the bottom row of Figure 3. Firstly (left and right picture), the 2d joint position must be detected properly. The middle picture also rendered a problem, which according to the original paper comes down to poses being not included in the Human3.6M dataset, such as upside-down poses or examples not showing a full human body.

The authors conclude that basic neural networks today are already able to produce very good results in terms of accuracy for uplifting 2d to 3d human poses. Therefore one of the main error sources of this process remains 2d pose estimation and more complex models should be able to perform the task of

uplifting even better.

*2) SVMAC: Unsupervised 3D Human Pose Estimation from a Single Image with Single-view-multi-angle Consistency:* [19] proposes a generative adversarial network (GAN) method which works unsupervised with just 2d joint positions as inputs. The generator gets trained to lift the input to a 3d pose estimation and extract the camera position, which enables rotating the model to reproject it into two dimensions. The reprojected 2d joint positions aswell as the ground truth ones are used to train the corresponding discriminator of the GAN. Single-view-multi-angle Consistency (SVMAC) denotes the ability to mix different rotations of an estimate pose with those of the estimated camera and ensuring that reprojections into 2d from the same angle are consistent. This is used in the loss function of the generator. The structure of the network is shown in Figure 4. SVMAC constraints are applied using the generators on the right. SVMAC resulted in heavily improved performance of the model when using just 2 angles. However using more than 2 did not yield a significantly better result but increased training time by a lot. Another experiment done was including 5% ground truth data (3d poses) for supervision (with 2 angles used for SVMAC), which led to impressive results being able to outperform many weakly- and even some fully supervised approaches in terms of PMPJPE (see subsection III-F). When considering only unsupervised methods, this one outperformed any previous one.

### B. 3d pose directly from 2d image

Methods for 3d human pose estimation directly from an imagine (or many) show the lack of 3d training data not captured by special indoor motion capture systems. This often leads to worse performance when comparing to uplifting methods in a real scenario test. If more in-the-wild datasets are coming up in the future, generating a 3d pose directly from a 2d image could be outperforming the uplifting approach. For now, generating artificial or weakly annotated training data or using multitask networks that can estimate 2d and 3d poses yields the best results in this area [112].

*1) Structured Prediction of 3D Human Pose with Deep Neural Networks:* Tekin et al. [100] showed that using a separately trained autoencoder to apply knowledge about human poses and combining it with a deep neural network can improve prediction accuracy. First, the autoencoder was trained to extract poses which were prepared with gaussian noise. This resulted in the autoencoder learning constraints about human poses. The high dimensional middle layer was assumed to contain most of that information, leading to the actual CNN being trained to output this layer for the corresponding input picture. Once this training was completed, Tekin et al. added the decoding layers (from middle layer to 3d model) of the autoencoder to the network, such that the output now was an actual human pose. After some fine-tuning of the now complete CNN experiments could be conducted. They showed that this approach led to an improvement when compared to previous methods, however their results can not keep up with uplifting methods such as presented in subsubsection VI-A1. The authors conclude that the framework used is generic and therefore can be used in future projects and applied to other prediction problems.

*2) 2D/3D Pose Estimation and Action Recognition using Multitask Deep Learning:* In [60] a convolutional deep neural network structure is proposed, which is able to handle 2d and 3d pose estimation from still images plus action recognition from video, since those problems are coupled. While not achieving groundbreakingly new state-of-the-art performance results in any of the solved tasks individually, this approach goes to show that a single architecture is able to produce at least equally as good results as dedicated methods for each task separately. Since the solved problems overlap, the network can work very efficient computationally. The authors claim that merging the tasks and training them together also increases robustness. Datasets used include Human3.6M [41] and MPII Human Poses [2], which render comparison with the approach presented in subsubsection VI-A1 possible. The multitask convolutional neural network outperforms [65] quite a lot, what in turn proves that complex network structures can produce even better results, as Martinez et al. stated.

### C. model-based 3d pose from 2d image

The approaches to model-based estimation of human posture are either optimization-based, where a model is iteratively refined from its initial state to fit the observed object, or regression-based, where deep neural networks extract features of the image to construct the model from shape and pose parameters. The former methods allow an exact match of image and model, but are not particularly fast and sensitive to the initialization. The latter provide acceptable but not entirely accurate results, are faster but depend on their supervision [46].
A common example of an optimization-based approach is found in the publication by Bogo et al. [5] from 2016. It is the first application of *SMPL* in the field of human pose estimation. The *SMPLify* algorithm is based on optimization and gradually tries to align the mean initial *SMPL* human model to 2d keypoints of the initial image.
However, *HMR* is a pose estimation network, which is a regression-based method published by Kanazawa et al. [44]. In their work, an encoder-regressor-discriminator architecture is discribed in which the encoder detects features that are regressed on pose, shape and camera parameters using an iterative regression module and finally passed on to the discriminator. The discriminator is learned from real data and has the purpose of detecting whether the parameters refer to real body meshes, so it acts as a supervisor.
A recent pose estimation approach is presented by Rong et al. [89] which starts from the concept of splitting the estimation of human pose into hand, face and total body, which are then merged into a final model. Each aspect utilises its own regressors, which are trained on datasets selected for the respective purpose. In accordance with this separation of concerns, the different network architectures and approaches are explained in the further sections.

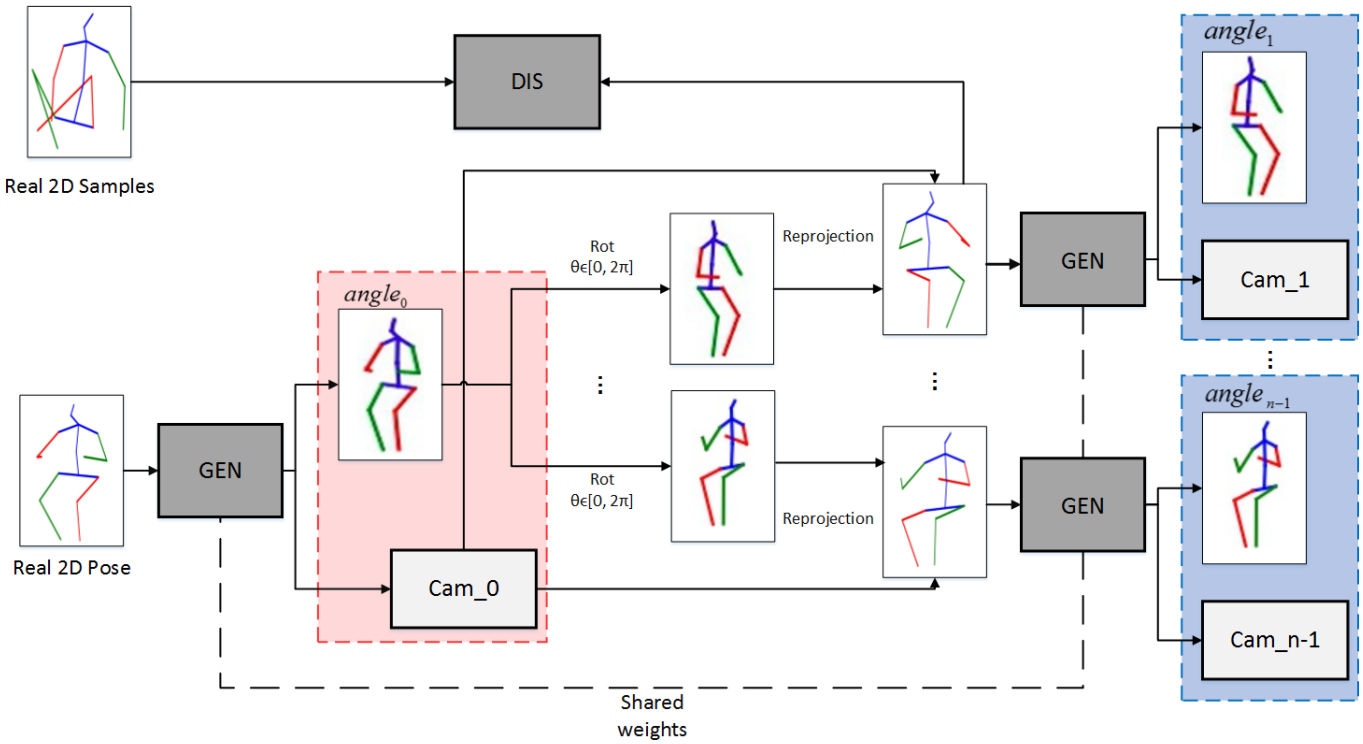


Fig. 4. GAN structure from [19]. The input 2d pose gets uplifted to a 3d pose estimation and an estimated camera position by the generator. This allows rotation of the model to feed the discriminator with a reprojection from various angles, who is comparing them with real 2d poses. The reprojections are also used to uplift and reproject them again to ensure Single-view-multi-angle Consistency.

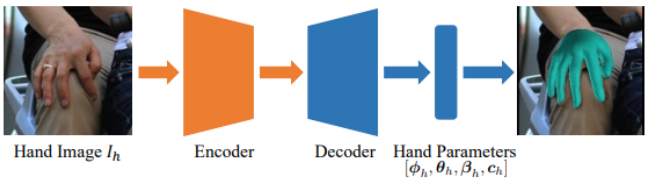


Fig. 5. Encoder-decoder architecture that takes a hand image as input, extracts the features and decodes them as sufficient parameters to construct a hand model. [89]

*1) hand estimation:* In *FrankMocap* by Rong et al. [89] the estimation of human hand posture is based on regression and uses a similar structure to the *HMR* shown in Figure 5 with an ecoder-decoder structure. Since *SMPL* does not support detailed hand shapes and postures, FrankMocap uses *SMPL-X* to include this information, which extends SMPL from subsection IV-A to include hand movements and facial expressions. Thus, the parameters $[\phi_h, \theta_h, \beta_h, c_h]$, i.e. the global orientation, pose and shape data of the hands, as well as the camera parameters were calculated. For the network, the annotations in the trained dataset are axis-angle representations to indicate 3D positions and 3d or 2d vertices as common positions, and therefore each annotation has its unique loss function. To increase the robustness of the model, data augmentation techniques such as random scaling, rotation, etc. and a blur filter are applied to the dataset to be learned.

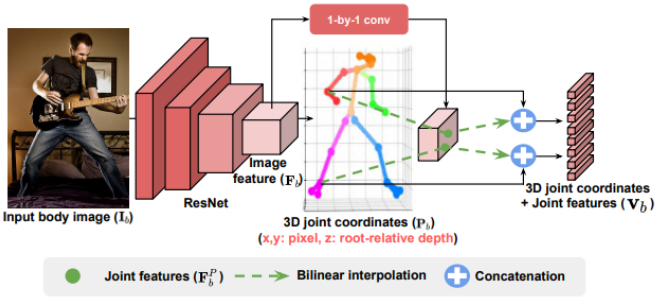


Fig. 6. Documentation of the *Pose2Pose* structure applied to the body (b). First $ResNet$ calculates the features of an input, which were used to calculate the joint positions and features with 1-by-1 convolutional layers. The features and positions were merged as input to the final layer of $ResNet$. [69]

Choutas et al. [17] contributed with their approach *ExPose* to overcome the limitations imposed by the small pixel area covered by a person's hands and face. In neural networks the resolution becomes even smaller, resulting in fewer relevant pixels in these areas and thus less meaningful data. Therefore, *body-driven attention* is introduced as a procedure in which the low-resolution body joints are first calculated from the pose parameters $\theta_b$ in order to recognise the hand and face areas in

the high-resolution image. Thus, a bounding box is calculated from the corresponding hand and face joints in the low-resolution image and projected onto the high-resolution image. Finally, the high-resolution hand and face images were fed into the respective neural networks, to refine the specific initial low-resolution pose parameters $\theta_b^{wrist}, \theta_b^{fingers}, \theta_b^{face}, \psi_b$ by adding predicted offsets. Thus, the wrist, hand and face posture as well as the expression coefficients can be represented by Equation 9.

$$\begin{aligned}[\theta^{wrist}, \theta^{finger}] &= [\theta_b^{wrist}, \theta_b^{finger}] + [\Delta\theta_{wrist}, \Delta\theta_{finger}] \\ [\psi, \theta^{face}] &= [\psi_b, \theta_b^{face}] + [\Delta\psi, \Delta\theta_{face}]\end{aligned} \tag{9}$$

Another paper by Moon et al. [69] states that the estimation of the hand depends on the prediction of the rotation of the wrist and fingers and the accuracy of the results relies on two crucial aspects. Firstly, considering the human kinematic chain, the wrist is connected to the metacarpophalangeal *MCP* joints, the finger roots. Knowledge of the *MCP* joints therefore benefits the calculation of the wrist and thus the estimation of the hand. In addition, the body pose can be used to determine whether the wrist rotation is anatomically correct. Lastly, the finger rotations were a product of body and hand features, which contained unnecessary information about the body and background, as well as rough information about the hand due to its small size.

To this end, *Pose2Pose* is described as a 3d joint position guided framework for predicting 3d joint rotations. It extracts the $J$ 3d joint positions $P$ from an input image through the neural network *ResNet* that encodes x and y in pixel space, while z is relative to the root joint, the wrist. The result is a feature map $F$ from which headmaps $H$ are obtained by applying a 1-by-1 convolutional layer, reducing the channel dimension of $F$ to $8 \cdot J$. Finally, the common locations are obtained by reshaping $H$ and calculating the *soft-argmax* operation [96].

In order to calculate the pose parameters, joint feature information is gained by performing positional pose-guided pooling *PPP*. For this purpose, a specific 1-by-1 convolution is applied to reduce the image dimensions, followed by a bilinear interpolation between the layer output and $P$, without considering the z-axis. The outcome of this practice are the joint features that, concatenated with $P$ and inserted into the final fully-connected layer of *ResNet*, yield the desired pose parameters. Figure 6 shows the *Pose2Pose* system applied to the body, which is also used for the hands in the approach by Moon et al.

*2) full-body pose estimation:* Similar to hand estimation, shape $\beta_{reg}$, pose $\theta_{reg}$ and camera parameters $Pi_{reg}$ can also be regressed in whole body pose estimation approaches. The supervision is communly processed by projecting keypoint information between dimensionens using the optained camera parameters, to use them in loss functions. An example of such a 2d loss function is given in [12] by Dongyue et al. where the ground truth joint positions are directly regressed

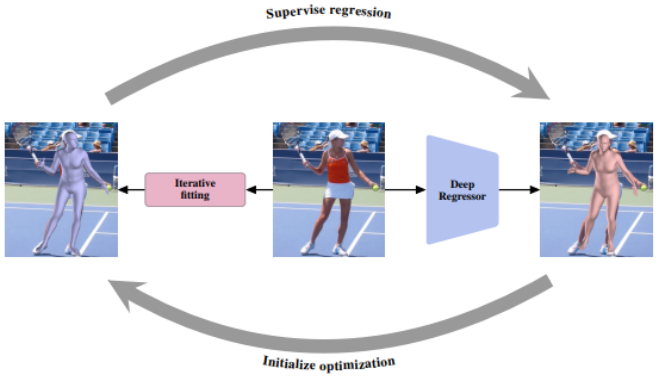


Fig. 7. Visualisation of the $SPIN$ loop, in which first the pose and shape parameters on the right side were regressed to initialize the $SMPLify$ algorithm on the left side. The optimized parameters from $SMPLify$ were used to improve the supervision of the regressor modul, resulting into better initial models for $SMPLify$. [46]

from the neural network *AlphaPose*, which are then compared to the associated joint positions resulting from the 3d-to-2d projections of the pose parameters. On the other hand, the regressed 3 dimensional information about pose and shape from *HMR* are used as ground truth in 3d loss functions. As in the work of Dongyue et al, pose estimation is done by initializing the pose and shape parameters using *HMR* regression and minimize the 2d loss function from which new camera parameters are calculated to finally minimize a total loss function which is the sum of the 2d, 3d and face losses. The results of this process are the pose and shape parameters. The algorithm *SPIN* [46] used in *FrankMocap* goes further by combining the advantages of optimization and regression-based implementations by using *SMPLify* and neural networks in loops. Substituting $\beta_{reg}$ and $\theta_{reg}$ into Equation 4 yields a human output mesh $M(\theta_{reg}, \beta_{reg})$. Instead of projecting the 3d joint positions into 2d space through a pre-trained regressor based on the camera information $\Pi_{reg}$, which is then used in the loss function for supervision, *SPIN* bypasses this process by calculating optimised poses $\theta_{opt}$ and shape $\beta_{opt}$ parameters through *SMPLify* and uses them as pseudo ground truth in a 3d loss function. This self-impruving nature of *SPIN* can be observet in a small example.

In the first iteration, the parameters $[\theta_{reg}, \beta_{reg}]$ are regressed by a neural network, which is used to initialise the stepwise optimisation process *SMPLify*. After its last step, both the pose and shape parameters are refined $[\theta_{opt}, \beta_{opt}]$, resulting in an optimized body model $M(\theta_{reg}, \beta_{reg})$, which is then used as ground truth in the loss function to optimize supervision of the regression module. In further iterations, *SMPLify* is instantiated with a better initial model from the regressed parameters, resulting in faster and more accurate computation. The loop is also showed in Figure 7.

However, all networks, including those for the body shape and pose parameters in the work of Choutas et al. [17], the process of *HMR* is followed.

Finally, Moon et al. [69] intend to include the *MCP* joint information as it is beneficial for body estimation. Thus, in its overall human pose estimation structure called *Hand4Whole*, the hand poses were detected prior to the body. The *MCP* joint features and positions can be additionally included in the posture estimation algorithm by passing them as additional values to the last phase of *pose2pose*, which are then inserted into *ResNet* network.

*3) face estimation:* As with human bodies and hands, faces also require parameters for *linear blend skinning* approaches that can be regressed from neural networks. The *FLAME* model, for instance, relies on shape $\beta_f$ and pose $\theta_f$ parameters as well as expression parameters $\psi$. Thus Equation 4 of the *SMPL* model and $T_F$ in Equation 6 can be reformulated, which is viewed in Equation 10. As with *SMPL*, pose, shape and additionally expression blend shapes [P,S,E] must be learned. [53]

$$\begin{aligned} M(\vec{\beta}_f, \vec{\theta}_f, \vec{\psi}) &= W(T_F(\vec{\beta}_f, \vec{\theta}_f, \vec{\psi}), J(\vec{\beta}_f), \vec{\theta}_f, \mathcal{W}) \\ T_F(\vec{\beta}_f, \vec{\theta}_f, \vec{\psi}) &= T + B_S(\vec{\beta}) + B_P(\vec{\theta}_f) + B_E(\vec{\psi}) \end{aligned} \tag{10}$$

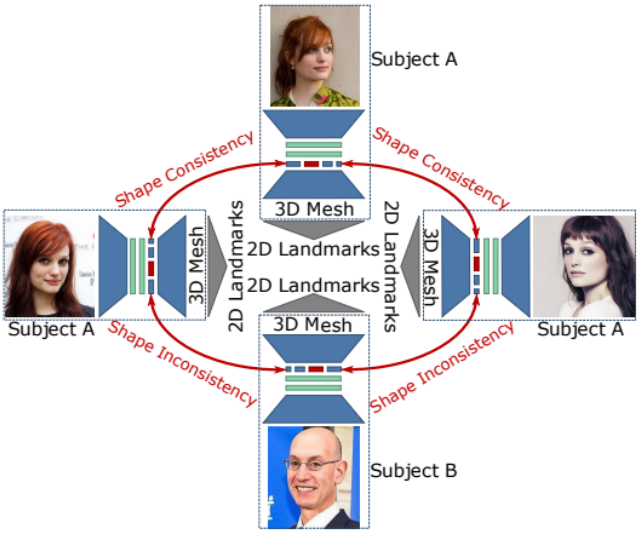


Fig. 8. Ring-shaped network architecture of the $RingNet$ in training with 4 elements that are encoder-decoder structures. Training is performed with image sources consisting of 2 subjects, with only one image coming from the second subject. Ring instances regress 3d information from 2d images, which are projected into 2d space for evaluation in the loss function. This concept ensures the learning of consistency. [90]

*RingNet* [90] is a regresson based approach to get the necessery parameters for the *FLAME* model. It introduces a ring-shaped architecture to increase the consistency of a person's face shape across different images. During the training, the 3d landmarks of a face were calculated and projected into 2d space to compare them in the loss function with the real landmarks of the dataset, which is displayed in Figure 8. To learn consistency, a dataset provides many images of the same subject, which are then learned in a ring structure with an image of another person. Each element of the ring consists of an encoder-decoder structure, with encoders sharing weights with other encoders. An encoder calculates high-dimensional features that are iteratively regressed by the encoder's regression network in an error feedback loop. Distinctness, which consists of producing the same shapes for the same individual but different shapes for different people, is achieved by including a threshold $\eta$ in the loss function, which implies that the distance between equal subjects is always $\eta$ smaller than the distance between unequal individuals.

### D. Direct 3d pose estimation

Methods working directly on 3D data, such as [116] (working with depth maps) can avoid potential sources of error such as projections or lighting conditions. This leads to more robust and accurate results, however more complex (neural network) architectures and computational resources are required. Traditional methods like the least-squares-estimation presented in [32] work without training data needed and are computationally inexpensive, but yield rather high error rates compared to newer methods.

## VII. Conclusion

As research in the topic of human pose estimation is increasing, it becomes more important to have an overview about methods used in this field. Therefore, this paper presented selected works of the last years. First commonly used datasets were described, followed by metrics for benchmarking to be able to compare different methods against each other. After that, as a representative of models in (3d) human pose estimation, SMPL was introduced in subsection IV-A.
2d pose estimation was next. Since neural networks are currently the most powerful tools in this area, challenges when designing those have to be discussed first. The top-down approach, which divides input images such that only a single person appears on one part, was now further investigated. While heat-map-based approaches are the most precise at the moment, regression-based methods with convolutional neural networks are more efficient in scenarios where real-time performance is important, because they are not as computationally expensive compared to the many steps a heat-map method needs. Also, training is easier with regression-based methods, as it can be done in an end-to-end fashion. Unlike top-down methods, bottom-up approaches try to estimate joint positions from an input image first, and then merge them to individual humans. Since the pose is not calculated for each human individual, these methods are faster than their top-down complement for the most part when multiple humans can be found in an image, because the performance of top-down approaches drops with the number of human poses to be estimated increasing. However, if not many persons are to be found (and they dont overlap each other), top-down methods

yield higher performance. Future research needs to be focussed to improve computational speed even further to make real-time applications possible, and decrease failure rates in scenarios with many humans or occlusion.

In 3d human pose estimation, the lack of 3d datasets of humans in-the-wild for training and testing neural networks should be addressed in the future, because just like in 2d human pose estimation, neural networks currently play a huge role in this field. Uplifting an already estimated pose in two dimensions to the third one has been one of the recent approaches in the past years to make up for the missing training data. However, this method relies heavily on the correctness of the 2d estimation, as it can not correct it. Performance-wise, the uplifting process does not need to be very complex computationally, which was shown by Martinez et al. in [65]. A different method, which used multiple reprojections of the generated 3d pose estimation to 2d to employ Single-view-multi-angle Consistency, was described too as a representative for unsupervised uplifting methods. The next approach, estimating a 3d pose directly from a 2d image was discussed. The get rid of the problem of missing in-the-wild training data, the methods presented here used a multitask network for 2d and 3d pose estimation or a previously seperately trained autoencoder. Working directly on 3d data yields high accuracy and robustness in exchange for high computational complexity and demand for a lot of computational resources. Since 3d data is not available in most applications in the near, this approach is not as important for future research though. Lastly, methods working with human 3d models, such as the previously described SMPL, were presented.

## REFERENCES


[1] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

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

[3] Mykhaylo Andriluka, Stefan Roth, and Bernt Schiele. Monocular 3d pose estimation and tracking by detection. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 623–630, 2010.

[4] Bruno Artacho and Andreas Savakis. UniPose: Unified Human Pose Estimation in Single Images and Videos. Technical Report arXiv:2001.08095, arXiv, January 2020. arXiv:2001.08095 [cs] type: article.

[5] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J. Black. Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In *Computer Vision – ECCV 2016*, Lecture Notes in Computer Science, pages 561–578. Springer International Publishing, October 2016.

[6] Adrian Bulat and Georgios Tzimiropoulos. Human pose estimation via Convolutional Part Heatmap Regression. volume 9911, pages 717–732. 2016. arXiv:1609.01743 [cs].

[7] Yuanhao Cai, Zhicheng Wang, Zhengxiong Luo, Binyi Yin, Angang Du, Haoqian Wang, Xiangyu Zhang, Xinyu Zhou, Erjin Zhou, and Jian Sun. Learning Delicate Local Representations for Multi-Person Pose Estimation. Technical Report arXiv:2003.04030, arXiv, July 2020. arXiv:2003.04030 [cs] type: article.

[8] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. Technical Report arXiv:1611.08050, arXiv, April 2017. arXiv:1611.08050 [cs] type: article.

[9] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-End Object Detection with Transformers. Technical Report arXiv:2005.12872, arXiv, May 2020. arXiv:2005.12872 [cs] type: article.

[10] Joao Carreira, Pulkit Agrawal, Katerina Fragkiadaki, and Jitendra Malik. Human Pose Estimation with Iterative Error Feedback. Technical Report arXiv:1507.06550, arXiv, June 2016. arXiv:1507.06550 [cs] type: article.

[11] Ching-Hang Chen and Deva Ramanan. 3d human pose estimation = 2d pose estimation + matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[12] Dongyue Chen, Yuanyuan Song, Fangzheng Liang, Teng Ma, Xiaoming Zhu, and Tong Jia. 3d human body reconstruction based on smpl model. *The Visual Computer*, 2022.

[13] Wenzheng Chen, Huan Wang, Yangyan Li, Hao Su, Zhenhua Wang, Changhe Tu, Dani Lischinski, Daniel Cohen-Or, and Baoquan Chen. Synthesizing training images for boosting human 3d pose estimation. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 479–488, 2016.

[14] Xianjie Chen and Alan Yuille. Articulated Pose Estimation by a Graphical Model with Image Dependent Pairwise Relations. Technical Report arXiv:1407.3399, arXiv, November 2014. arXiv:1407.3399 [cs] type: article.

[15] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded Pyramid Network for Multi-Person Pose Estimation. Technical Report arXiv:1711.07319, arXiv, April 2018. arXiv:1711.07319 [cs] type: article.

[16] Bowen Cheng, Bin Xiao, Jingdong Wang, Honghui Shi, Thomas S. Huang, and Lei Zhang. HigherHRNet: Scale-Aware Representation Learning for Bottom-Up Human Pose Estimation. Technical Report arXiv:1908.10357, arXiv, March 2020. arXiv:1908.10357 [cs, eess] type: article.

[17] Vasileios Choutas, Georgios Pavlakos, Timo Bolkart, Dimitrios Tzionas, and Michael J. Black. Monocular expressive body regression through body-driven attention. In *European Conference on Computer Vision (ECCV)*, pages 20–40, 2020.

[18] Xiao Chu, Wei Yang, Wanli Ouyang, Cheng Ma, Alan L. Yuille, and Xiaogang Wang. Multi-Context Attention for Human Pose Estimation. Technical Report arXiv:1702.07432, arXiv, February 2017. arXiv:1702.07432 [cs] type: article.

[19] Yicheng Deng, Yongqi Sun, and Jiahui Zhu. SVMA: A gan-based model for monocular 3d human pose estimation. *CoRR*, abs/2106.05616, 2021.

[20] Junting Dong, Wen Jiang, Qixing Huang, Hujun Bao, and Xiaowei Zhou. Fast and robust multi-person 3d pose estimation from multiple views. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[21] M. Eichner and V. Ferrari. Human Pose Co-Estimation and Applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012.

[22] Marcin Eichner, Manuel Marin-Jimenez, Andrew Zisserman, and Vittorio Ferrari. 2d articulated human pose estimation and retrieval in (almost) unconstrained still images. *International journal of computer vision*, 99(2):190–214, 2012.

[23] Xiaochuan Fan, Kang Zheng, Yuewei Lin, and Song Wang. Combining Local Appearance and Holistic View: Dual-Source Deep Neural Networks for Human Pose Estimation. Technical Report arXiv:1504.07159, arXiv, April 2015. arXiv:1504.07159 [cs] type: article.

[24] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Pictorial Structures for Object Recognition. *International Journal of Computer Vision*, 61(1):55–79, January 2005.

[25] Feng Ning, D. Delhomme, Y. LeCun, F. Piano, L. Bottou, and P.E. Barbano. Toward automatic phenotyping of developing embryos from videos. *IEEE Transactions on Image Processing*, 14(9):1360–1371, September 2005.

[26] Mihai Fieraru, Anna Khoreva, Leonid Pishchulin, and Bernt Schiele. Learning to Refine Human Pose Estimation. Technical Report arXiv:1804.07909, arXiv, April 2018. arXiv:1804.07909 [cs] type: article.

[27] Zigang Geng, Ke Sun, Bin Xiao, Zhaoxiang Zhang, and Jingdong Wang. Bottom-Up Human Pose Estimation via Disentangled Keypoint Regression. pages 14676–14686, 2021.

[28] Rohit Girdhar, Georgia Gkioxari, Lorenzo Torresani, Manohar Paluri, and Du Tran. Detect-and-Track: Efficient Pose Estimation in Videos. Technical Report arXiv:1712.09184, arXiv, May 2018. arXiv:1712.09184 [cs] type: article.

[29] Georgia Gkioxari, Alexander Toshev, and Navdeep Jaitly. Chained Predictions Using Convolutional Neural Networks. Technical Report arXiv:1605.02346, arXiv, October 2016. arXiv:1605.02346 [cs] type: article.

[30] John C Gower. Generalized procrustes analysis. *Psychometrika*, 40(1):33–51, 1975.

[31] Ikhsanul Habibie, Weipeng Xu, Dushyant Mehta, Gerard Pons-Moll, and Christian Theobalt. In the Wild Human Pose Estimation Using Explicit 2D Features and Intermediate 3D Representations. pages 10905–10914, 2019.

[32] R.M. Haralick, H. Joo, C. Lee, X. Zhuang, V.G. Vaidya, and M.B. Kim. Pose estimation from corresponding point data. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(6):1426–1446, 1989.

[33] Mohamed Hassan, Vasileios Choutas, Dimitrios Tzionas, and Michael J. Black. Resolving 3D human pose ambiguities with 3D scene constraints. In *Proceedings International Conference on Computer Vision*, pages 2282–2292. IEEE, October 2019.

[34] Yana Hasson, Gül Varol, Dimitrios Tzionas, Igor Kalevatykh, Michael J. Black, Ivan Laptev, and Cordelia Schmid. Learning joint reconstruction of hands and manipulated objects. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 11807–11816, June 2019.

[35] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[36] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, June 2016. ISSN: 1063-6919.

[37] David T. Hoffmann, Dimitrios Tzionas, Michael J. Black, and Siyu Tang. Learning to train with synthetic humans. In *German Conference on Pattern Recognition (GCPR)*, pages 609–623. Springer International Publishing, September 2019.

[38] Yinghao Huang, Manuel Kaufmann, Emre Aksan, Michael J. Black, Otmar Hilliges, and Gerard Pons-Moll. Deep inertial poser: Learning to reconstruct human pose from sparse inertial measurements in real time. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 37:185:1–185:15, November 2018. Two first authors contributed equally.

[39] Eldar Insafutdinov, Mykhaylo Andriluka, Leonid Pishchulin, Siyu Tang, Evgeny Levinkov, Bjoern Andres, and Bernt Schiele. ArtTrack: Articulated Multi-Person Tracking in the Wild. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1293–1301, Honolulu, HI, July 2017. IEEE.

[40] Eldar Insafutdinov, Leonid Pishchulin, Bjoern Andres, Mykhaylo Andriluka, and Bernt Schiele. DeeperCut: A Deeper, Stronger, and Faster Multi-person Pose Estimation Model. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, Lecture Notes in Computer Science, pages 34–50, Cham, 2016. Springer International Publishing.

[41] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1325–1339, 2014.

[42] Sheng Jin, Wentao Liu, Wanli Ouyang, and Chen Qian. Multi-person Articulated Tracking with Spatial and Temporal Embeddings. Technical Report arXiv:1903.09214, arXiv, March 2019. arXiv:1903.09214 [cs] type: article.

[43] Sheng Jin, Wentao Liu, Enze Xie, Wenhai Wang, Chen Qian, Wanli Ouyang, and Ping Luo. Differentiable Hierarchical Graph Grouping for Multi-person Pose Estimation. In *Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII*, pages 718–734, Berlin, Heidelberg, August 2020. Springer-Verlag.

[44] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7122–7131. IEEE Computer Society, 2018.

[45] Lipeng Ke, Ming-Ching Chang, Honggang Qi, and Siwei Lyu. Multi-Scale Structure-Aware Network for Human Pose Estimation. Technical Report arXiv:1803.09894, arXiv, September 2018. arXiv:1803.09894 [cs] type: article.

[46] Nikos Kolotouros, Georgios Pavlakos, Michael J. Black, and Kostas Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[47] Sven Kreiss, Lorenzo Bertoni, and Alexandre Alahi. PifPaf: Composite Fields for Human Pose Estimation. pages 11977–11986, 2019.

[48] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

[49] Jia Li, Wen Su, and Zengfu Wang. Simple Pose: Rethinking and Improving a Bottom-up Approach for Multi-Person Pose Estimation. Technical Report arXiv:1911.10529, arXiv, November 2019. arXiv:1911.10529 [cs] type: article.

[50] Ke Li, Shijie Wang, Xiang Zhang, Yifan Xu, Weijian Xu, and Zhuowen Tu. Pose Recognition with Cascade Transformers. Technical Report arXiv:2104.06976, arXiv, April 2021. arXiv:2104.06976 [cs] type: article.

[51] S. Z. Li. Markov Random Field Models in Computer Vision, 1994.

[52] Sijin Li, Zhi-Qiang Liu, and Antoni B. Chan. Heterogeneous Multi-task Learning for Human Pose Estimation with Deep Convolutional Neural Network. Technical Report arXiv:1406.3474, arXiv, June 2014. arXiv:1406.3474 [cs] type: article.

[53] Tianye Li, Timo Bolkart, Michael J. Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4D scans. *ACM Transactions on Graphics*, 36(6):194:1–194:17, November 2017. Two first authors contributed equally.

[54] Ita Lifshitz, Ethan Fetaya, and Shimon Ullman. Human Pose Estimation using Deep Consensus Voting. Technical Report arXiv:1603.08212, arXiv, March 2016. arXiv:1603.08212 [cs] type: article.

[55] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2014.

[56] Wentao Liu, Jie Chen, Cheng Li, Chen Qian, Xiao Chu, and Xiaolin Hu. A Cascaded Inception of Inception Network With Attention Modulated Feature Fusion for Human Pose Estimation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), April 2018. Number: 1.

[57] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, October 2015.

[58] Yue Luo, Jimmy Ren, Zhouxia Wang, Wenxiu Sun, Jinshan Pan, Jianbo Liu, Jiahao Pang, and Liang Lin. LSTM Pose Machines. pages 5207–5215, 2018.

[59] Zhengxiong Luo, Zhicheng Wang, Yan Huang, Liang Wang, Tieniu Tan, and Erjin Zhou. Rethinking the Heatmap Regression for Bottom-up Human Pose Estimation. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13259–13268, Nashville, TN, USA, June 2021. IEEE.

[60] Diogo C. Luvizon, David Picard, and Hedi Tabia. 2d/3d pose estimation and action recognition using multitask deep learning. *CoRR*, abs/1802.09232, 2018.

[61] Qianli Ma, Jinlong Yang, Anurag Ranjan, Sergi Pujades, Gerard Pons-Moll, Siyu Tang, and Michael J. Black. Learning to dress 3D people in generative clothing. In *Computer Vision and Pattern Recognition (CVPR)*, pages 6468–6477. IEEE, June 2020.

[62] Qianli Ma, Jinlong Yang, Siyu Tang, and Michael J. Black. The power of points for modeling humans in clothing. In *Proc. International Conference on Computer Vision (ICCV)*, pages 10974–10984, October 2021.

[63] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. AMASS: Archive of motion capture as surface shapes. In *Proceedings International Conference on Computer Vision*, pages 5442–5451. IEEE, October 2019.

[64] Gines Hidalgo Martinez, Yaadhav Raaj, Haroon Idrees, Donglai Xiang, Hanbyul Joo, Tomas Simon, and Yaser Sheikh. Single-Network Whole-Body Pose Estimation. In *2019 IEEE/CVF International Conference*

*on Computer Vision (ICCV)*, pages 6981–6990, Seoul, Korea (South), October 2019. IEEE.

[65] Julieta Martinez, Rayat Hossain, Javier Romero, and James J. Little. A simple yet effective baseline for 3d human pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[66] Antonio S. Micilotta, Eng-Jon Ong, and Richard Bowden. Real-Time Upper Body Detection and 3D Pose Estimation in Monoscopic Images. In Aleš Leonardis, Horst Bischof, and Axel Pinz, editors, *Computer Vision – ECCV 2006*, Lecture Notes in Computer Science, pages 139–150, Berlin, Heidelberg, 2006. Springer.

[67] Thomas B. Moeslund. *Interacting with a Virtual World Through Motion Capture*, pages 221–234. Springer London, London, 2001.

[68] Thomas B. Moeslund and Erik Granum. A survey of computer vision-based human motion capture. *Computer Vision and Image Understanding*, 81(3):231–268, 2001.

[69] Gyeongsik Moon, Hongsuk Choi, and Kyoung Mu Lee. Accurate 3d hand pose estimation for whole-body 3d human mesh estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 2308–2317, June 2022.

[70] Lea Müller, Ahmed A. A. Osman, Siyu Tang, Chun-Hao P. Huang, and Michael J. Black. On self-contact and human pose. In *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 9990–9999, June 2021.

[71] Alejandro Newell, Zhiao Huang, and Jia Deng. Associative Embedding: End-to-End Learning for Joint Detection and Grouping. Technical Report arXiv:1611.05424, arXiv, June 2017. arXiv:1611.05424 [cs] type: article.

[72] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked Hourglass Networks for Human Pose Estimation. Technical Report arXiv:1603.06937, arXiv, July 2016. arXiv:1603.06937 [cs] type: article.

[73] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation, 2016.

[74] Xuecheng Nie, Jiashi Feng, Junliang Xing, and Shuicheng Yan. Pose Partition Networks for Multi-person Pose Estimation. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, volume 11209, pages 705–720. Springer International Publishing, Cham, 2018. Series Title: Lecture Notes in Computer Science.

[75] Xuecheng Nie, Jiashi Feng, Jianfeng Zhang, and Shuicheng Yan. Single-Stage Multi-Person Pose Machines. pages 6951–6960, 2019.

[76] Xuecheng Nie, Jiashi Feng, Yiming Zuo, and Shuicheng Yan. Human Pose Estimation With Parsing Induced Learner. pages 2100–2108, 2018.

[77] Sungheon Park, Jihye Hwang, and Nojun Kwak. 3D Human Pose Estimation Using Convolutional Neural Networks with 2D Pose Information. In Gang Hua and Hervé Jégou, editors, *Computer Vision – ECCV 2016 Workshops*, Lecture Notes in Computer Science, pages 156–169, Cham, 2016. Springer International Publishing.

[78] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G. Derpanis, and Kostas Daniilidis. Coarse-to-fine volumetric prediction for single-image 3d human pose. *CoRR*, abs/1611.07828, 2016.

[79] Tomas Pfister, James Charles, and Andrew Zisserman. Flowing ConvNets for Human Pose Estimation in Videos. Technical Report arXiv:1506.02897, arXiv, November 2015. arXiv:1506.02897 [cs] type: article.

[80] Leonid Pishchulin, Eldar Insafutdinov, Siyu Tang, Bjoern Andres, Mykhaylo Andriluka, Peter Gehler, and Bernt Schiele. DeepCut: Joint Subset Partition and Labeling for Multi Person Pose Estimation. Technical Report arXiv:1511.06645, arXiv, April 2016. arXiv:1511.06645 [cs] type: article.

[81] Lingteng Qiu, Xuanye Zhang, Yanran Li, Guanbin Li, Xiaojun Wu, Zixiang Xiong, Xiaoguang Han, and Shuguang Cui. Peeking into occluded joints: A novel framework for crowd pose estimation. Technical Report arXiv:2003.10506, arXiv, March 2020. arXiv:2003.10506 [cs] type: article.

[82] Umer Rafi, Bastian Leibe, Juergen Gall, and Ilya Kostrikov. An Efficient Convolutional Network for Human Pose Estimation. In *Procedings of the British Machine Vision Conference 2016*, pages 109.1–109.11, York, UK, 2016. British Machine Vision Association.

[83] Varun Ramakrishna, Daniel Munoz, Martial Hebert, James Andrew Bagnell, and Yaser Sheikh. Pose Machines: Articulated Pose Estimation via Inference Machines. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, Lecture Notes in Computer Science, pages 33–47, Cham, 2014. Springer International Publishing.

[84] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. Technical Report arXiv:1506.01497, arXiv, January 2016. arXiv:1506.01497 [cs] type: article.

[85] Helge Rhodin, Jörg Spörri, Isinsu Katircioglu, Victor Constantin, Frédéric Meyer, Erich Müller, Mathieu Salzmann, and Pascal Fua. Learning monocular 3d human pose estimation from multi-view images. *CoRR*, abs/1803.04775, 2018.

[86] K.M. Robinette, H. Daanen, and E. Paquet. The caesar project: a 3-d surface anthropometry survey. In *Second International Conference on 3-D Digital Imaging and Modeling (Cat. No.PR00062)*, pages 380–386, 1999.

[87] Gregory Rogez and Cordelia Schmid. Mocap-guided data augmentation for 3d pose estimation in the wild. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

[88] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6):245:1–245:17, November 2017.

[89] Yu Rong, Takaaki Shiratori, and Hanbyul Joo. Frankmocap: A monocular 3d whole-body pose estimation system via regression and integration. In *IEEE International Conference on Computer Vision Workshops*, 2021.

[90] Soubhik Sanyal, Timo Bolkart, Haiwen Feng, and Michael Black. Learning to regress 3d face shape and expression from an image without 3d supervision. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[91] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014.

[92] Kai Su, Dongdong Yu, Zhenqi Xu, Xin Geng, and Changhu Wang. Multi-Person Pose Estimation with Enhanced Channel-wise and Spatial Information. Technical Report arXiv:1905.03466, arXiv, May 2019. arXiv:1905.03466 [cs] type: article.

[93] Shigemasa Sumi. Upside-down presentation of the johansson moving light-spot pattern. *Perception*, 13(3):283–286, 1984. PMID: 6514513.

[94] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep High-Resolution Representation Learning for Human Pose Estimation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5686–5696, Long Beach, CA, USA, June 2019. IEEE.

[95] Xiao Sun, Jiaxiang Shang, Shuang Liang, and Yichen Wei. Compositional Human Pose Regression. Technical Report arXiv:1704.00159, arXiv, August 2017. arXiv:1704.00159 [cs] type: article.

[96] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression, 2017.

[97] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral Human Pose Regression. Technical Report arXiv:1711.08229, arXiv, September 2018. arXiv:1711.08229 [cs] type: article.

[98] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going Deeper with Convolutions. Technical Report arXiv:1409.4842, arXiv, September 2014. arXiv:1409.4842 [cs] type: article.

[99] Omid Taheri, Nima Ghorbani, Michael J. Black, and Dimitrios Tzionas. Grab: A dataset of whole-body human grasping of objects. In *Computer Vision – ECCV 2020*, volume LNCS 12355, pages 581–600, Cham, August 2020. Springer International Publishing.

[100] Bugra Tekin, Isinsu Katircioglu, Mathieu Salzmann, Vincent Lepetit, and Pascal Fua. Structured prediction of 3d human pose with deep neural networks. *CoRR*, abs/1605.05180, 2016.

[101] Bugra Tekin, Pablo Marquez-Neila, Mathieu Salzmann, and Pascal Fua. Learning to Fuse 2D and 3D Image Cues for Monocular Body Pose Estimation. pages 3941–3950, 2017.

[102] Zhi Tian, Hao Chen, and Chunhua Shen. DirectPose: Direct End-to-End Multi-Person Pose Estimation. Technical Report arXiv:1911.07451, arXiv, November 2019. arXiv:1911.07451 [cs] type: article.

[103] Denis Tome, Chris Russell, and Lourdes Agapito. Lifting from the deep: Convolutional 3d pose estimation from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[104] Jonathan Tompson, Ross Goroshin, Arjun Jain, Yann LeCun, and Christopher Bregler. Efficient Object Localization Using Convolutional Networks. Technical Report arXiv:1411.4280, arXiv, June 2015. arXiv:1411.4280 [cs] type: article.

[105] Jonathan Tompson, Arjun Jain, Yann LeCun, and Christoph Bregler. Joint Training of a Convolutional Network and a Graphical Model for Human Pose Estimation. Technical Report arXiv:1406.2984, arXiv, September 2014. arXiv:1406.2984 [cs] type: article.

[106] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[107] Alexander Toshev and Christian Szegedy. DeepPose: Human Pose Estimation via Deep Neural Networks. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1653–1660, June 2014. arXiv:1312.4659 [cs].

[108] Gül Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J. Black, Ivan Laptev, and Cordelia Schmid. Learning from synthetic humans. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2017*, pages 4627–4635, Piscataway, NJ, USA, July 2017. IEEE.

[109] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. Technical Report arXiv:1706.03762, arXiv, December 2017. arXiv:1706.03762 [cs] type: article.

[110] Timo von Marcard, Roberto Henschel, Michael J. Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3D human pose in the wild using IMUs and a moving camera. In *European Conference on Computer Vision (ECCV)*, volume Lecture Notes in Computer Science, vol 11214, pages 614–631. Springer, Cham, September 2018.

[111] Jian Wang, Xiang Long, Yuan Gao, Errui Ding, and Shilei Wen. Graph-PCNN: Two Stage Human Pose Estimation with Graph Pose Refinement. Technical Report arXiv:2007.10599, arXiv, July 2020. arXiv:2007.10599 [cs] type: article.

[112] Jinbao Wang, Shujie Tan, Xiantong Zhen, Shuo Xu, Feng Zheng, Zhenyu He, and Ling Shao. Deep 3D human pose estimation: A review. *Computer Vision and Image Understanding*, 210:103225, September 2021.

[113] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional Pose Machines. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4724–4732, Las Vegas, NV, USA, June 2016. IEEE.

[114] Wei Yang, Shuang Li, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang. Learning Feature Pyramids for Human Pose Estimation. Technical Report arXiv:1708.01101, arXiv, August 2017. arXiv:1708.01101 [cs] type: article.

[115] Yi Yang and Deva Ramanan. Articulated human detection with flexible mixtures of parts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12):2878–2890, 2013.

[116] Mao Ye, Xianwang Wang, Ruigang Yang, Liu Ren, and Marc Pollefeys. Accurate 3d pose estimation from a single depth image. In *2011 International Conference on Computer Vision*, pages 731–738, 2011.

[117] Yixing Gao, Hyung Jin Chang, and Yiannis Demiris. User modelling for personalised dressing assistance by humanoid robots. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1840–1845, Hamburg, Germany, September 2015. IEEE.

[118] Ce Zheng, Wenhan Wu, Taojiannan Yang, Sijie Zhu, Chen Chen, Ruixu Liu, Ju Shen, Nasser Kehtarnavaz, and Mubarak Shah. Deep learning-based human pose estimation: A survey. *CoRR*, abs/2012.13392, 2020.

[119] Kun Zhou, Xiaoguang Han, Nianjuan Jiang, Kui Jia, and Jiangbo Lu. HEMlets Pose: Learning Part-Centric Heatmap Triplets for Accurate 3D Human Pose Estimation. pages 2344–2353, 2019.

[120] Xiangyu Zhu, Yingying Jiang, and Zhenbo Luo. Multi-Person Pose Estimation for PoseTrack with Enhanced Part Affinity Fields. page 4.

[121] Silvia Zuffi, O. Freifeld, and Michael Black. From Pictorial Structures to Deformable Structures. June 2012.