A Review of Human Pose Estimation and Tracking

Anonymous ACCV 2018 submission

Paper ID ***

Abstract. Human pose estimation is an important problem that aims to estimate the body part or joint positions of a person from an image or a video. The problem has several variations including the number of people to be tracked, the type of inputs provided (RGB, depth, infrared etc.), the number of cameras available, whether the input is an image or a video sequence, and if the joint estimation is to be done in two dimensions (2D) or three dimensions (3D). There have been a few surveys over the years that focus on a singular subdomain. This paper aims to provide a comprehensive review of recent pose estimation literature covering all these variations. We begin by defining the problem of pose estimation and describing the various methodologies used for tackling this problem including various types of body models, features as well as deep learning approaches for different types of problems like static single person pose estimation, static multi person pose estimation, pose estimation in videos and pose estimation using multiple cameras. We describe the different datasets and metrics used in literature and then provide a provide a short description of the algorithms used by various papers along with their results on these datasets. Finally we compare these different approaches and give some directions for future research.

1 Introduction

Human pose estimation and tracking is an important problem in computer vision [1] that is receiving increased attention from researchers in the past few years [1–3]. Pose estimation aims to estimate the location and/or orientation of body parts or joints of one or more humans from an image or video sequence [1]. This is useful for many application such as medical gait analysis, smart surveillance systems, character animation in movies or games [4], and human-computer interaction, among other applications.

Traditionally, pose estimation is done using optical markers attached to the body, which can be used to accurately recover the layout of the body. However, these methods are expensive and invasive. Markerless approaches, on the other hand, capture a scene using one or more RGB cameras or depth-based cameras such as the Microsoft Kinect [5], and use the captured images to estimate the pose algorithmically. Advances in deep learning and availability of large labeled datasets and increased computational power has made it possible to estimate the pose reasonably accurately from the image data without requiring expensive cameras or elaborate body markers. Deep learning based markerless approaches are inexpensive and non-invasive and their accuracy has been steadily improving, due to which they can potentially be used for many more applications than marker-based approaches.

The pose estimation problem can be categorized based on the types of inputs and the complexity of outputs along the following five axes: (a) number of people being being tracked, (b) static still image or dynamic video, (c) number of cameras used to capture images, (d) types of images (RGB, depth, infrared) acquired and (e) complexity of the output body model. Depending on the above combinations of types of inputs and outputs there can be a large variety of pose estimation problems and algorithms suitable to solve them. In Section 2, we begin by defining the pose estimation problem.

Any pose estimation algorithm has the following three important components: (a) the body model that is used to map the human body using an abstract mathematical representation, (b) a set of features derived from the input data of RGB and/or depth image or video and (c) the actual algorithm used to map these features into the mathematical representation of the body model. The body models, categories of features used and the types of algorithms used for solving the pose estimation problem are described in detail in Section 3.

Most of the techniques used for pose estimation are based on machine learning or deep learning based algorithms. These algorithms require significant data to train their models and evaluate their performance. Moreover, in order to effectively compare different approaches, one needs to evaluate them on the same dataset and use the same metric for performance. In order to facilitate this research, several open access datasets have been made available. In Section 4, we review the datasets and performance metrics used by these papers to train their models and evaluate their performance.

Finally, in Section 5, we review the approaches that have been used to solve different variants of pose estimation problems. Papers were obtained from SCOPUS by searching between the time period of 2011-2018 using the keywords 'human', 'pose', 'pose estimation' and 'tracking' followed by manually filtering out irrelevant papers. Some seminal papers from outside this time period have been included for completeness. We also focus on key ideas and approaches that can be generalized to cover other problems in this domain.

In Section 6, we conclude with a discussion on the currently successful approaches and scope for future work in the domain of pose estimation.

2 Pose Estimation: Problem Definition

Pose estimation involves the identification of joints/body parts of one or more humans from an image or video sequence. We can classify the problem of human pose estimation along the following five axes:

Number of people being tracked: The pose estimation algorithm can either track a single person or multiple people in the scene of an image or video

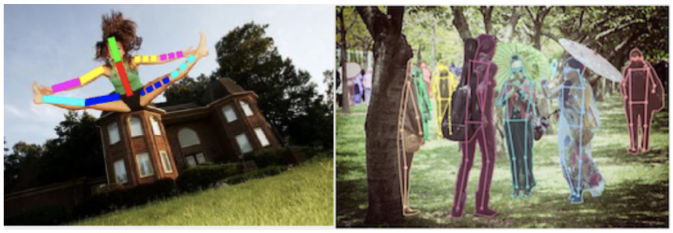

Fig. 1. (a) Example of joint positions for one person, taken from LSP dataset[6], (b) Example of joint positions for multiple people, taken from COCO dataset[7]

sequence. Multi-person pose estimation needs to handle the additional problem of inter-person occlusion and may require additional preprocessing to obtain bounding boxes of each person which may not be needed in the single person case. Figure 1(a) shows the input image of one person along with its corresponding skeleton. Figure 1(b) shows the multi-person case. Initial approaches of pose estimation were largely focused on single person pose estimation, however with the availability of big multi-person datasets such as MPII[8] and COCO[7], the multi person problem has lately been getting increased attention.

Types of Images: Most of the pose estimation techniques use a Red-Green-Blue (RGB) image as input. In order to improve the accuracy and to possibly output the joint coordinates in three dimensions, some researchers have also employed depth images. Additionally, infra-red (IR) images are available from low cost devices like the Kinect that may be used to improve accuracy, though very few researchers have used IR information for pose estimation. Additionally, one may consider hyperspectral imaging technology[9] to improve accuracy, though such cameras are not currently used for pose estimation. If multiple types of images are used for pose estimation, they need to be registered to each other. In some cases, such as in the Kinect, the image acquisition system may provide the registered images. If registered images are not available, an additional software based registration step is required during pre-processing.

Static vs Dynamic Pose Estimation: For the static problem, the pose needs to be estimated using a still image where as in the dynamic case, a series of poses need to be produced for an input video sequence. For the dynamic problem, the estimated poses should ideally be consistent across successive frames of video, the arrival and departure of people dynamically entering or leaving the scene needs to be taken care of, and the algorithm needs to be computationally efficient to handle large number of frames on which the problem is to be solved. Solving the occlusion problem might be easier in the case of dynamic pose estimation due to the availability of past or future frames where the body part is not occluded. It is possible to apply static pose estimation techniques for the dynamic problem by breaking it down as multiple independent static problems. However, in practice, results are generally not as good as expected due to jitter and inconsistency problems [4].

Number of viewpoints/cameras: The most common research work solves the pose estimation problem using a monocular input (ie. a single camera). In the case of multiple people, or when 3D output is required, data from multiple viewpoints may be combined to generate more accurate point cloud information and handle occlusions more effectively. The research on multi-camera pose estimation is currently somewhat limited [10–13]. It may be possible to get very accurate 3D pose using multiple cameras in the future. More research is needed in this direction.

Richness of body model: Most techniques use a simple N-joint rigid skeleton model in 2-dimensions (N is typically between 13 to 30). Such a model is well suited for Human Computer Interaction (HCI) applications. However for many industrial applications a more elaborate model may be needed. Certain techniques produce 3D skeleton coordinates using depth data as input or RGB data alone. More elaborate human skeleton models may be required for applications in medical or animation domains. One may consider a highly detailed mesh model such as in [14], which may find many novel applications.

3 Methodologies for Human Pose Estimation

The methodologies used by different techniques for human pose estimation can be generally abstracted into the following five steps:

(a) **Preprocessing** which may involve background removal, creation and tracking of a bounding box for each person tracked, and registration in case of multiple modalities or viewpoints.

(b) **Body modeling** which involves creating a model for the body including its various properties such as rigidity, elasticity, dimensionality etc.

(c) **Feature extraction** which involves extraction of key features from RGB, depth and other images/videos that are provided to the machine learning (ML) algorithm.

(d) **Learning and Inference** which comprise the machine learning and inference approach that converts the extracted features into the body parts model.

(e) **Post-processing** which involves processing the output of the machine learning model with the constraints of body model to give the final output.

3.1 Preprocessing

Background removal or background subtraction involves segmentation or separation of the human from the background. Monocular depth based approaches such as [15, 16] employ it as a preprocessing step. Popular algorithms for background subtraction include first frame subtraction, Codebook model, minimum subtraction and single Gaussian model. For a detailed comparison of background subtraction algorithms see [42].

Bounding box creation is typically employed by top-down pose estimation algorithms[17], which are detailed in section 3.4. It involves segmenting out each human being in the input image by creating a rectangular bounding box around each human. Object detection approaches such as R-CNN [18] and its successors are the most commonly used for bounding box creation.

Registration is required in case inputs are available from multiple cameras, either in the same modality or in multiple modalities (RGB, depth etc.). Registration requires a calibration step for each camera which is used to provide a transformation matrix than can then be used to map their input into a single set of world coordinates. A survey of camera calibration techniques is presented in [19]. Calibration algorithms are implemented in libraries like openCV [20] that can be used out of the box.

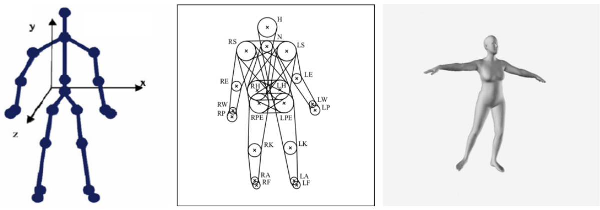

Fig. 2. (a) Image along with the corresponding color gradients, used to calculate HOG and similar features. Courtesy: https://www.learnopencv.com/histogram-of-oriented-gradients/ (b) Image with SIFT features

3.2 Body Models

A body model converts the abstract problem of human pose estimation into a concrete problem of estimation of certain parameters using the image data. A body is modeled as a set of connected body parts and for each body part parameters like its position in space and/or orientation need to be estimated. Prior constraints or distribution of these parameters may be specified in the body model. The body model may be a 2D or 3D representation of the human body. Based on our review, we have identified three major types of models used in literature - kinematic models, shape-based models and mesh-based models - which we formally define below. Fig. 2 shows an example of each kind of body model.

Kinematic: Formally, kinematic models can be represented as a graph $G = (V, E)$, where each vertex $v \in V$ represents a joint or a body part [21]. The pose estimation algorithms assign a position either in a 2D plane or in 3D space to each vertex of the body model. The edges in E encode constraints or prior beliefs about the structure of the body model. These priors or constraints can be over the distribution of edge lengths and over angles two adjacent edges can make. The edges typically describe the skeleton structure of the human body (though some models may have edges not necessarily corresponding to the bones, eg. between eyes and ears). This model can be incorporated in a Bayesian framework of learning. Some techniques may even learn the edges in the graph.

Shape-based: In shape-based models, human body parts are approximated using geometric shapes like rectangles, cylinders, conics etc. Shape-based models can be 2D or 3D. An example of a 2-dimensional shape-based model is the one used in [22], in which each body part is represented as a rectangle characterised by the average color and the color histogram. A 3-dimensional shape-based model is used in [23], in which each body part is represented as a conic consisting of the end points and radii.

Mesh-based: Meshes are models in which the body is represented as a point cloud, so each point on the surface of a body part is part of the model. Meshes are typically acquired using 3D scans of the human body. Meshes are the most expressive body models and can be used to capture non-trivial poses with high amount of detail. Examples of mesh models include Shape Completion and Animation of People (SCAPE)[24] and Skinned Multi-Person Linear Model (SMPL)[25].

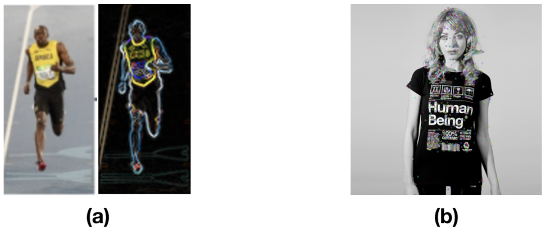

Fig. 3. Left to Right: Kinematic body model, Shape-based capsule model from [23], Mesh-based SMPL body model [25]

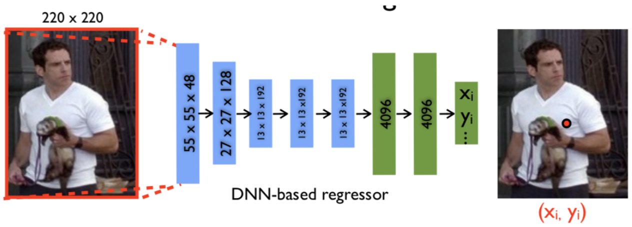

Fig. 4. The convolutional neural network proposed in [1], one of the first papers to use deep learning for pose estimation

3.3 Feature Extraction

Feature extraction refers to the creation of derived values from the raw data (such as an image or video) that can be used as input to a learning algorithm. Features can either be implicit or explicit.

3.3.1 Explicit Feature Extraction

Older approaches like [26, 27] use explicit features that are created manually. Some frequently used explicit features are detailed below.

Histogram of oriented gradients (HoG)[28]: This techniques calculates the color gradient magnitude and orientation in small patches of an image. Bins corresponding to the gradient orientation are created which become features are their values depend on the gradient magnitudes. Gradient magnitudes for an example image is shown in Fig 3(a) which shows HOG helps in detecting important features.

Scale Invariant Feature Transform (SIFT)[29]: As the name suggests, this is a feature that is invariant to scale. It analyses and extracts features that are present are multiple scales (resolution) of the image using successive convolutions of the image with a gaussian filter and finds features that are salient at multiples scales. SIFT features detected on a person are shown in Fig 3(b).

Edgelet/Shapelet features: These are explicit kernels used to detect certain edges or shapes in the input image.

Motion features: Motion features can be used to improve performance for pose estimation in videos by taking into account temporal information. The most commonly used motion feature is the dense optical flow. Optical flow is used to detect motion of an object or person is a scene by analysing the intensity gradient between successive frames, and can thus help in tracking. Simpler but higher dimensional features like the RGB difference image have also been used[30].

3.3.2 Implicit Feature Extraction

Newer deep learning based approaches like [1] use implicit features, which means they are not manually created but inferred during the learning process. The most popular way of learning implicit features is using convolutional neural networks (CNNs)[31].

Convolutional Neural Network (CNN): CNNs are variants of neural networks where the input image is successively convolved with multiple kernels, followed by a non-linearity to extract features. They typically also include max-pooling layers after each convolutional layer to reduce dimensionality and provide local invariance to translation and rotation.

3.4 Learning and Inference

3.4.1 Static Pose Estimation

Static pose estimation involves pose estimation on a static image. Algorithms for static pose estimation can be broadly classified into bottom up and top down approaches.

Bottom Up Approach: Bottom up approaches involve first detecting the parts or joints for one or humans in the image, and then assembling the parts together and associating them with a particular human. They can be classified into part-based and deep learning based approaches.

Part-based methods solve pose estimation problems by first detecting body part candidates in an image, and then connecting the body parts according to a body model. The most commonly used model is the Pictorial Structures Model (PSM) and its variants. A PSM is a type of kinematic model with nodes representing body parts and edges representing connections between body parts. Inference is usually performed in a Bayesian manner where the objective is maximize the probability of the pose estimate given the image evidence. Effectively this decomposes into unary potentials encoding the probability of a part configuration given image evidence and binary potentials encoding the probability of a body part configuration given an adjacent body part configuration. Parameters for the unary potentials are learnt from the training data using techniques like random forests, SVMs and gaussian mixture models. The binary potentials are typically gaussian whose parameters are learnt from the training data using maximum likelihood. Inference is performed using message passing or belief propagation and inference is exact and efficient due to the tree structure of the PSM. Some approaches which do not use a tree model employ approximate methods like loopy belief propagation.

Convolutional neural networks (CNNs) are the most popular and successful models for bottom-up pose estimation currently. In the context of bottom up pose estimation, CNNs are typically used to give a probability density (also called heat map) for each joint position, followed by techniques like integer linear programming (ILP) or bipartite matching to associate the joints to people.

Top Down Approach: Top Down approaches directly associate one human body to a pose estimate. In the case of multi-person pose estimation, these approaches typically involve a segmentation step where each human is first segmented into a bounding box, followed by pose estimation being performed individually on each human. Top down pose estimation can be classified into deep learning based and generative body-model based approaches.

Convolutional neural networks are the most recent and successful approach to top down estimation. Typically these are used both for the initial segmentation step as discussed earlier, as well as for directly predicting the keypoints for each segmented human. Techniques like [3] forego the segmentation step and instead use a modified version of Faster R-CNN to directly predict key-points instead of the bounding points.

Generative approaches model the likelihood of the observations given a pose estimate. The objective is to use a body model, typically a shape-based or mesh model, and fit the evidence to it. Inference involves a complex search over the state space to locate the peaks of the likelihood. Generative methods are susceptible to local minima, and thus require good initial pose estimates, regardless of the optimization scheme used. The pose is typically inferred using local optimization or stochastic search.

3.4.2 Tracking of Pose in Video

Tracking involves pose estimation on a video sequence (possibly in real time). It also involves two broad approaches - bottom up and top down.

Top Down Approach: Top down approaches incorporate the pose estimation approaches from bottom up or top down static pose estimation, but additionally they also incorporate priors from the pose estimate of previous frames. One example is [23], where a gaussian prior with zero mean and small variance is used to estimate possible deviation of a joint in the current frame from the previous frame. The intuition this uses is that estimate across successive frames should be consistent - the joints can only move a small distance from one frame to the next.

Bottom Up Approach: Bottom up approaches process each frame independently using the techniques from static pose estimation. However, as in [32], they use bipartite matching or integer linear programming to match the joints to different people across frames. The intuition here is that each person only moves a small distance from one frame to the next, which means their joints will be close to each other across frames. Some approaches, like [33], additionally use a 3D CNN to convolve along the time dimension as well.

3.4.3 Multi-camera pose estimation

[10] uses multiple kinect cameras for pose estimation. The main approach here is to register each camera and process images into a single set of world coordinates, which are then fed as inputs into a CNN using a top-down or bottom up approach.

Table 1. Pose Estimation Datasets

Name	No. of images	Type	Single/ Multi Person	2D/3D
Leeds Sports Pose (LSP) [34][6]	12000	Static	Single	2D
We are family [35]	525	Static	Multi	2D
SMMC-10 [36]	10 videos	Video	Single	3D
KTH Multiview Football I [37]	5907	Static	Single	2D
KTH Multiview Football II [38]	2000	Static	Single	3D
Frames Labeled in Cinema (FLIC) [39]	5003	Static	Single	2D
MPII Human Pose [8]	25000	Static	Single/Multi	2D
COCO Keypoints [7]	200000	Static	Multi	2D
HumanEva [40]	74267	Video	Single	3D
Human3.6M [41]	3.6M	Video	Single	3D
PoseTrack [42]	66374	Video	Multi	2D

4 Datasets and Metrics

4.1 Datasets

The creation of labeled datasets is one of the most important steps required for creation of accurate pose estimation techniques. Since most of the pose estimation techniques use machine learning and/or deep learning methods, the quality as well as size of labeled data is an extremely crucial factor in performance.

Early approaches used small, non-standardized datasets for training and evaluation. This limits the performance of the techniques and also makes it difficult to compare performance of different approaches [43]. In the last ten years, several large standardized labeled datasets have been made available for researchers. These datasets have been summarized in Table 1. Descriptions for the most important of these datasets have been provided below.

The Leeds Sports Pose (LSP) dataset[34] contains 2000 RGB images depicting people playing various sports collected from Flickr. Fourteen joints for each person covering the full body are annotated manually. Later, an extended LSP dataset[6] containing 10000 images collected and annotated using the same protocol was released. The Frames Labeled in Cinema (FLIC) dataset[39] contains 5003 RGB images taken from various Hollywood movies. One human in each image is annotated with ten upper body joints. Annotation was performed by five workers on Amazon Mechanical Turk (MTurk) per image. The MPII human pose dataset[8] is a dataset of over 25000 RGB images depicting nearly 500 different human activities collected from Youtube videos. Thirteen full body joints are annotated by a combination of experts and MTurk workers. Both the full single-person dataset and a multi-person subset of MPII dataset have been used for evaluation by various papers. The COCO keypoints dataset[7] is a multi-person dataset of over 200000 RGB images containing over 250000 people. Images are collected from Flickr, and eighteen full body joints for each person are annotated by MTurk workers.

Both COCO and MPII datasets share the advantage of being larger and much more diverse than the LSP and FLIC datasets. However, all four of these datasets only contain static images. PoseTrack[42] is a diverse multi-person video dataset containing around 550 RGB videos. Images in the MPII dataset were taken and 5 seconds of video around the image was annotated manually. Each person is annotated with 15 full body joints.

The above datasets contain joint annotations only in 2 dimensions. The most popular datasets providing joint positions in 3 dimensions are HumanEva and Human3.6M. HumanEva[40] contains video sequences recorded using multiple RGB and grayscale cameras. There are two subsets (HumanEva-I and II) which are recorded using different number of cameras. Ground truth 3D poses are captured using marker-based motion capture (mocap) cameras. There are a total of 4 subjects performing 6 common activities. Human3.6M [41] is a dataset in which 11 actors performing 15 different possible activities were recorded using RGB and time-of-flight (depth) cameras (4 each). 3D poses are obtained using 10 mocap cameras and 24 joints are annotated per person. While both HumanEva and Human3.6M datasets are very valuable as they provide 3D poses, they share the limitation of being single person datasets recorded in controlled environments, leading to lower variety.

4.2 Metrics

For the validation of human pose estimation algorithms, various metrics are used. The *percentage of correct parts* (PCP)[35] metric checks if the detected body part (comprising two end joints) is within some percentage (typically 50%) of the length of the ground truth body part. The *percentage of correct keypoints* (PCK)[44] metric checks if the distance between the detected keypoint and ground truth keypoint is within a fraction (typically 0.2) of the person's bounding box height/width. *Average precision of keypoint* (APK)[44] is similar to PCK but not widely used. *Percentage of detected joints* (PDJ) is also similar but instead of bounding box dimension it uses the torso diameter which is the distance between left hip and right shoulder. The *mean average precision* (mAP) calculates the average precision of joint detections. The PCP, PCK, APK and PDJ metrics can all be plotted on a curve by varying the percentage/fraction used to indicate correct detection.
The COCO keypoints dataset uses an evaluation measure based on distance between ground truth and detected keypoints called *object keypoint similarity* (OKS). Thresholding the OKS allows them to calculate *average precision* (AP) and *average recall* (AR). Finally, evaluations on 3D pose datasets like HumanEva and Human3.6M is done using the 3D error, also called *mean per-joint precision error* (MPJPE) which is the mean of the Euclidean distance between detected and ground truth keypoints. *Mean per joint angle error* (MPJAE) that considers joint angles and *Mean per joint localization error* (MPJLE) which binarizes the error by thresholding the distance (like in PCP) are also used[41].

Table 2. Types of pose estimation problems solved in recent literature

Paper	**Still/ Video**	**Single/Multi Person**	**Image Type**	Single/Multi Camera
Shotton et. al.	Static	Single	Depth	Single
Girschick et. al.	Static	Single	Depth	Single
Jung et. al.	Static	Single	Depth	Single
Newell et. al.	Static	Single	RGB	Single
Sapp et. al.	Static	Single	RGB	Single
Wei et. al.	Static	Single	RGB	Single
Chu et. al.	Static	Single	RGB	Single
Ning et. al.	Static	Single	RGB	Single
Cao et. al.	Static	Multi	RGB	Single
He et. al.	Static	Multi	RGB	Single
Pischulin et. al.	Static	Multi	RGB	Single
Fang et. al.	Static	Multi	RGB	Single
Xia et. al.	Static	Multi	RGB	Single
Ning et. al.	Static	Multi	RGB	Single
Newell et. al.	Static	Multi	RGB	Single
Guler et. al.	Static	Multi	RGB	Single
Ganapathi et. al.	Video	Single	Depth	Single
Pfister et. al.	Video	Single	RGB	Single
Jin et. al.	Video	Multi	RGB	Single
Insafutdinov et. al.	Video	Multi	RGB	Single
Girdhar et. al.	Video	Multi	RGB	Single
Daubney et. al.	Video	Single	RGB	Single
Song et. al.	Video	Multi	RGB	Single
Shafaei et. al.	Static	Single	Depth	Multi
Zhang et. al.	Static	Single	Depth	Multi
Phan et. al.	Static	Single	Depth	Multi
Elhayek et. al.	Static	Single	Depth	Multi

5 Review of different Algorithms for Pose Estimation

5.1 Static Single Person Pose Estimation

Static multi-person pose estimation means that a single image is provided as input and the joints of a single person in the image have to be detected. Sometimes the image may contain multiple people but detection is performed for the most prominent person.

Monocular depth-based pose estimation techniques: A single depth image is taken as input by [15, 16, 45], typically using a Kinect camera. Standard background removal techniques are first used to segment out the human. A random forest is employed by [15] to classify each pixel in the segmented image as belonging to one of 32 body parts. Simple depth features are computed - for each pixel, the difference in depth at an offset in two directions is taken as a feature for

Table 3. Results on COCO kepyoints dataset

Paper	AP	$\mathbf{AP}_{50}$	$\mathbf{AP}_{75}$	$\mathbf{AP}_M$	$\mathbf{AP}_L$
Cao et. al.	61.8	84.9	67.5	57.1	68.2
Papandreou et. al.	68.1	87.1	75.5	65.8	73.3
He et. al.	63.1	87.3	68.7	57.8	71.4
Chen et. al.	72.1	91.4	80.0	68.7	77.2
Fang et. al.	72.3	89.2	79.1	68.0	78.6
Newell et. al.	65.5	86.8	72.3	60.6	72.6

Table 4. Results of various papers on the MPII Multi Person Dataset

Paper	Hea	Sho	Elb	Wri	Hip	Kne	Ank	Total
Iqbal et. al.	58.4	53.9	44.5	35.0	42.4	36.7	31.1	43.1
DeeperCut	78.4	72.5	60.2	51.0	57.2	52.0	45.4	59.5
ArtTrack	88.8	87.0	81.4	72.5	77.7	73.0	68.1	79.7
Cao et. al.	91.2	87.6	77.7	66.8	75.4	68.9	61.7	75.6
Newell et. al.	92.1	89.3	78.9	69.8	76.2	71.6	64.7	77.5
Fang et. al.	91.3	90.5	84.0	76.4	80.3	79.9	72.4	82.1

the random forest. Different offset directions give different features. Finally, the body part pixels are clustered and the mean of each cluster is the joint position. The random forest is replaced with a regression forest in [16], which predicts how far the current pixel is from each joint. These distances are clustered to obtain each joint position. [45] also uses a regression forest but instead of storing the offset to a joint, they store the direction. Joint locations are obtained by performing random walks starting from various random positions. This approach is extremely fast and outperforms [15] and [16] on the SMMC-10 dataset.

Part-based models: Pose estimation using a single RGB image as input is performed in [46, 44, 47]. A Pictorial Structures Model (PSM)[21] is employed by [46] as their body model. A Bayesian framework is used for inference in which sum of unary potentials indicating the probability of the part detection and binary potentials indicating the likelihood of adjacent part locations is maximized using tree based message passing. Parameters for unary potentials are learnt using boosted random forests while those of binary potentials are learnt using maximum likelihood. The PSM model is enhanced using a mixture of parts model in [44], where the pose probability is maximized over several mixture components. A structural SVM is used for learning and message passing along with non-maximum suppression for inference. A richer model than PSM is also used by [47] which increases its expressiveness but preserves the tree structure by introducing latent variables representing higher level body parts like left arm, right arm, torso etc. with finer body parts as children of these latent nodes. A type variable for each body part is also introduced (that can represent an open or closed palm for example) and maximization is performed over mixture of types as well as pose. Parameters are learnt using a latent SVM. An additional

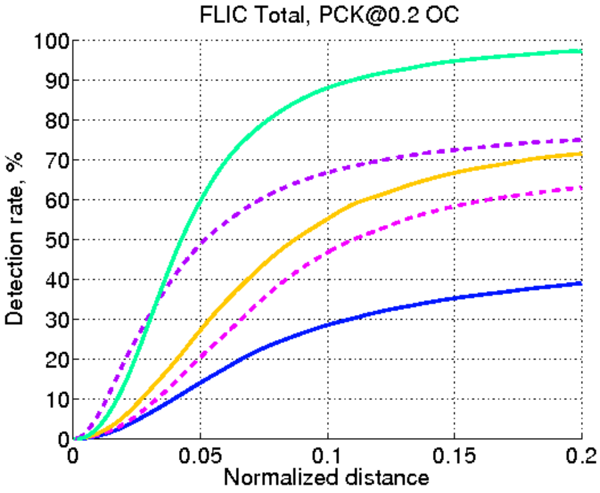

Fig. 5. Results of various papers on FLIC dataset

intermediate representation called poselets is used by [48], that can represent clusters of body parts. The PSM is conditioned on these detected poselets. Multiple body models are employed by [39] - a pictorial structures model (PSM) with different shape parameters along with a deformable parts model (DPM). A convex objective is used to fit their model to one or more of the reference body models for each input. However, only upper body poses are predicted in [39]. All these models predominantly employ HOG features for part detections. Evaluations on LSP show that [48] outperformed the other approaches while [39] showed superior performance on the FLIC dataset that they introduced.

Deep Learning Models: A single RGB input is taken by [1, 49–52] as well. A CNN model is used to predict a set of joint positions given the image. The first deep learning model for human pose estimation was proposed by [1]. A modified version of the Alexnet model [31] is use that directly regress to joint locations. A cascade of such CNN models is used where each subsequent CNN is provided as input a bounding box image around the previously predicted joint, and this is done for each joint. At each stage the CNN refines the previous detection. A top-down hourglass shaped CNN called Stacked Hourglass Network is proposed in [49]. The CNN processes the image at multiple scales. Convolutional and max pooling layers are used to process features down to a very low resolution. At each max pooling step, the network branches off and applies more convolutions at the original pre-pooled resolution. After reaching the lowest resolution, the network begins the top-down sequence of upsampling and combination of features across scales. A similar approach as [1] is used in [50] but with a deeper CNN structure that predicts probability or heat maps for each joint. These probability maps are provided as input to cascading CNNs to refine the output at each step. The CNN

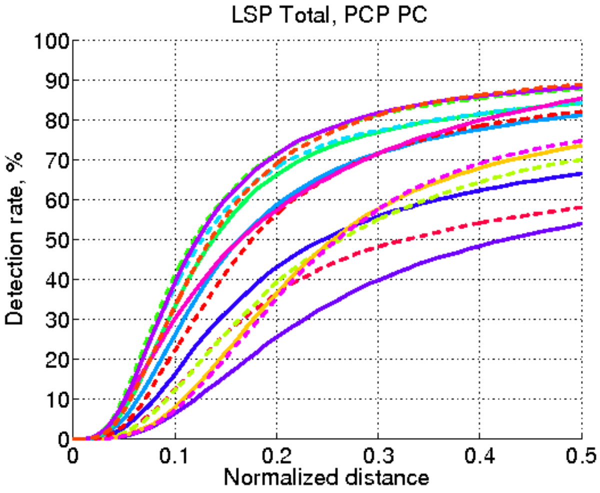

Fig. 6. Results of various papers on LSP dataset

model is enhanced with an attention mechanism [53] in [51] which focuses on relevant parts of the image for each joint at multiple resolutions. A Conditional Random Field (CRF) is utilized to model the correlations among neighboring regions in the attention map. The ResNet [54] model is adapted into their own units which have a larger receptive field. The Stacked Hourglass Network is combined with the Inception ResNet[55] model to produce joint predictions in [52]. The loss function incorporates external knowledge that takes the form of soft constraints that are imposed on the proposed joint model, based on shape, deformation and proximity. All the above techniques are evaluated on one or more of the LSP, FLIC and MPII single person datasets and show increasingly better performance.

5.2 Static Multi Person Pose Estimation

Static multi-person pose estimation means that a single image is provided as input and the joints of each person in the image have to be detected. This problem has been explored only for RGB inputs so every technique discussed takes as input a single RGB image.

Bottom Up Deep Learning Techniques: Bottom up pose estimation is a technique in which the joints in the image are detected first and then the joints are assembled and associated to each person. It is employed in [56, 57, 2]. A modification of Faster R-CNN[58], which is an object detection model, is used in [56] to give body part proposals. An ILP is formulated using constraints based on proximity of parts belonging to each person. Solving the ILP gives the skeleton for each human. A ResNet model, which is deeper and more accurate, is used by [57] for body part proposal. Some additional ILP constraints over [56] are added to improve performance. [2] improves on this by predicting heat maps for each joint using a model similar to [50]. Additionally for each pixel a vector (part affinity field) is generated which points along the direction of the body part it belongs to. Finally using the predicted joints and part affinity fields a bipartite matching is performed to generate the skeleton for each human. [2] was the winner of the COCO Keypoints 2016 challenge and also outperformed its predecessors on the MPII multi person dataset.

Top Down Deep Learning Techniques: A top down approach typically involves segmentation of each human in the image into a bounding box followed by single person pose estimation for each bounding box. This approach is used in [3, 17, 59]. An object detection model is proposed in [3] in which a single CNN has different branches that produce the bounding boxes and region proposals for different objects. For joint prediction in humans they use exactly the same pipeline but predicted the joint locations instead of the bounding box corners. A Spatial Transformer Network (STN) followed by a Spatial Detransformer Network (SDTN) is used by [17] to produce bounding boxes for each human. An additional STN (without the SDTN) is used for regularization of the prediction problem. For each bounding box, the approach of [1] is used to predict keypoints, and to eliminate redundant predictions for a joint, some constraints based on part shape and proximity of joints are added. A slightly modified version of the Stacked Hourglass Network is used by [59] that considers multiple detected peaks for each joint (in order to do multi person pose estimation). In addition to a joint prediction, the network simultaneously predicts a tag which associates the joint to a person. The loss function consists of the error in the predicted joint as well as the error in predicting the tag for each joint. Top-down approaches have increased in popularity and are currently state of the art on the COCO keypoints and MPII multi-person datasets.

Miscellaneous or hybrid approaches: Two Fully Connected Networks (FCNs) are employed by [60], one for predicting the joint probabilities (per pixel) and one for predicting the part probabilities for each human. To give the final prediction, a fully connected CRF with is used with potentials between connected joints, between connected parts and between part and joint, which induces a spatial consistency in the prediction. A dual path network is used by [61], where one path uses a top-down DenseNet for producing bounding box followed by keypoint detection and another path uses a bottom up Pose Machine with part affinity fields. The two paths are fused with another Dense Net to provide a final detection. The task of mapping the pixels of a person in a 2D RGB image to a corresponding 3D body model is tackled in [14]. A model based on Mask R-CNN is trained that first classifies each pixel into a body part and then regresses to the exact mapping. The COCO dataset was enhanced manually by partially mapping pixels to a body model for training, and this dataset will be released publicly for the community[14].

5.3 Tracking of Pose in Video

Tracking of pose in video means performing pose estimation a sequence of input images, possibly in real time.

Pose estimation on depth videos: A single depth image per time point is taken as input by [23]. A body model made of conics is used with each conic represented by two joints at the end and a radius. A Bayesian model is used comprising of a measurement model to predict the probability of input image given current joint prediction and a motion model to predict probability of current joint prediction given the previous joint prediction. A simple normal distribution is used to model the latter probability and a ray-constrained ICP approach is used to model the former probability. They additionally use constraints on the body model based on shape and scale compared to the reference model.

Single-person pose estimation on RGB videos: Single person pose estimation is performed by [62, 63, 30]. A CNN is used to predict the heat maps for each joint. In addition to the input image, motion features like the difference image and optical flow image are provided as input in [30]. Heat maps per joint for the current frame are also predicted in [62]. Then the heat maps produced from the previous N time points are aligned to the current one using dense optical flow. A parametric pooling layer is used to combine all the warped frames into a single detection using a 1D convolution. Dense optical flow for aligning adjacent heat maps is also employed by [63]. However, instead of pooling the heat maps, inference using loopy belief propagation is employed to minimize an objective function combining the detection error per frame and temporal consistency between frames.

Multi-person pose estimation in RGB videos Multi-person pose estimation on unconstrained RGB videos is tackled in [64, 65, 32, 33, 63]. A bottom-up approach is employed by [64] in which heat maps for joints in each frame are first generated using the same model as [57]. A spatio-temporal graph is constructed with edges within and across frames and an ILP solver is used to associated joints to each human. This helps provide temporally consistent detections. A similar approach is employed by [65] but they experiment with the models of [2] and [3] to obtain the initial joint predictions. A temporal graph modeled as a graph multicut problem is used by [32] Body part proposals for each individual frame are generated using two methods: bottom up and bottom up/top down. The bottom up approach directly predicts heat maps for each joint for all people. The top down/bottom up approach first predicts the head joint for each person and then uses that as a conditional to generate the rest of the joints for each person. A modified ResNet is used for both approaches. A top-down approach is employed by [33]. Instead of a temporal graph, a 3D extension of the Mask R-CNN[3] is used to handle the temporal information. The 2D CNN is replaced with a 3D CNN which takes as input 20 frames of video at a time. The most popular evaluation dataset for this task is the recent PoseTrack dataset, with state of the art results being obtained by [33]. [66] take as input an RGB video. They introduced a method that permits greater uncertainty in the root node of the probabilistic graph that represents a human body by stochastically tracking the root node and estimating the posterior over the remaining parts after applying temporal diffusion. The state of each node, excluding the root, is represented as a quaternion rotation and all the distributions are modeled as Gaussians. Inference is performed by passing messages between nodes and the Maximum-a-Posteriori (MAP) estimated pose is selected.

5.4 Multi-Camera Pose Estimation

Multi-Camera pose estimation means that the input image is acquired from multiple viewpoints using two or more RGB and/or depth cameras. A preprocessing step common to all approaches used for this task is calibration of each camera so that the images obtained from them can be registered to a single set of world co-ordinates.

Two or more depth images are taken as input by [10, 11] using multiple Kinect cameras. A Bayesian approach for multi-camera pose tracking is employed by [11], consisting of a motion model based on a Gaussian delta from the previous predicted pose and an observation model which seeks to minimize the distance of each observed point from the predicted pose using a kinematic body model. More recently, a CNN is used [10] to classify each pixel in each depth image as a body part after background subtraction. A single point cloud for the person is then obtained by combining the different images together using position and joint prediction features fed into a linear regressor. Finally they cluster the body parts pixels to obtain joint locations, similar to [15]. [67] directly use the skeleton poses provided by multiple Kinects and combine them using Kalman filters.

Both RGB and depth images from four Kinect cameras are used by [12]. They employ a shape-base body model composed of capsules and create a geodesic distance graph (GDG) which is a graphical representation of the human point cloud, and optimization on this graph gives the required pose. In order to incorporate information from previous frames to help with occlusion, optical flow information from each of the four previous RGB images is used along with a voting scheme.

[13] take as input an RGB image. They proposed a method for tracking the 3D human joints in both indoor and outdoor scenarios using as low as two or three cameras. For each joint in 2D, a discriminative part-based method was selected which estimates the unary potentials by employing a convolutional network. Pose constraints are probabilistically extracted for tracking, by using the unary potentials and a weighted sampling from a pose posterior guided by the model. These constraints are combined with a similarity term, which measures for the images of each camera, the overlap between a 3D model, and the 2D Sums of Gaussians (SoG) images. They evaluate their results on a custom MPI-MARCOnI containing 12 scenes.

6 Discussion and Future Directions

From the previous section we can clearly see that the trend for all kinds of pose estimation is towards deep learning and specifically, convolutional neural networks, due to their superior performance across tasks and datasets. For the purposes of discussion and comparison, we will first compare non-deep learning models like the earlier part-based models, regression models etc. followed by a comparison among different deep learning models. Finally we will provide a broad comparison between deep learning and non-deep learning methods, followed by possible future directions.

While both papers use a random forest for the task, Girschick et. al. predict the displacement of current pixel from joint rather than classifying a joint to a body part as done by Shotton et. al. This allows the model to more robustly predict the location of the joint, rather than a simple clustering of body parts. Moreover the features used are more directly suited to the task of Girchick et. al. Jung et. al. use a random walk to find a joint location, rather than clustering over possible displacements from a pixel. This does not improve performance much but allows huge computational gains, letting them do inference at 1000 frames per second. All of the above papers use Kinect devices for testing, which points out an obvious shortcoming, which is not using the RGB modality as an additional input. Incorporating RGB would help tackle some problems like occlusion (in some cases) and figuring out whether the subject is facing away from the camera. Shifaei et. al. use the same concept as Shotton et. al. but use deep convolutional networks instead of random forests to classify pixels into body parts along with combining predictions from multiple cameras.

Deep learning methods for RGB sequences have followed two tracks like we have described before - top down and bottom up. The main difference between various papers that follow one of these approaches is the architecture. The general trend for convolutional architectures has been towards increased depth and more receptive fields per layer, along with incorporating newer regularization techniques like batch normalization and layer normalization. There have also been structural innovations like the stacked hourglass network which processed the image at multiple scales and then uses deconvolutions to upscale the processed image. There is also a trend towards repeated refinement of the prediction, in which the prediction from the convolutional network is again provided to it as input to refine the prediction as seen in [2]. Comparison between bottom up and top down approaches also shows no clear winner, though the latest top-down models are state of the art by a small margin.

The main reason for the success of deep learning methods over earlier feautre-based approaches mirrors why it is successful in other domains: Deep models can learn features on their own and have a large capacity, along with availability of large amounts of training data and computational power, especially with the advent of the COCO and Human3.6M datasets.

Despite the recent successes in the field, there are still many challenges that need to be addressed. The main unsolved challenge in pose estimation is the problem of occlusion. While end to end deep learning approaches are better at handling occlusion than earlier parts-based approaches (despite not explicitly handling it), their main failure case is when some body parts are occluded by others (or other people). This is a bigger challenge when trying to estimate 3D pose, because for 2D pose estimation it is often OK not to detect occluded body parts. Another problem is when cloth colour closely resembles the background - this can lead to certain body parts like arms/legs not being detected. This problem can be handled better by having an explicit body model or taking into account depth information if available (like from a Kinect device). For temporal sequences, many approaches try to predict pose frame by frame, which leads to a lot of jitter. These challenges mean that pose estimation and tracking is still an open problem with a lot of opportunities for improvement.

7 Conclusion

References

1. Toshev, A., Szegedy, C.: Deeppose: Human pose estimation via deep neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2014) 1653–1660
2. Cao, Z., Simon, T., Wei, S.E., Sheikh, Y.: Realtime multi-person 2d pose estimation using part affinity fields. In: CVPR. Volume 1. (2017) 7
3. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Computer Vision (ICCV), 2017 IEEE International Conference on, IEEE (2017) 2980–2988
4. Mehta, D., Sridhar, S., Sotnychenko, O., Rhodin, H., Shafiei, M., Seidel, H.P., Xu, W., Casas, D., Theobalt, C.: Vnect: Real-time 3d human pose estimation with a single rgb camera. ACM Transactions on Graphics (TOG) **36** (2017) 44
5. Sell, J., O'Connor, P.: The xbox one system on a chip and kinect sensor. IEEE Micro **34** (2014) 44–53
6. Johnson, S., Everingham, M.: Learning effective human pose estimation from inaccurate annotation. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. (2011)
7. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV. (2014)
8. Andriluka, M., Pishchulin, L., Gehler, P., Schiele, B.: 2d human pose estimation: New benchmark and state of the art analysis. In: Proceedings of the IEEE Conference on computer Vision and Pattern Recognition. (2014) 3686–3693
9. Grahn, H., Geladi, P.: Techniques and applications of hyperspectral image analysis. John Wiley & Sons (2007)
10. Shafaei, A., Little, J.J.: Real-time human motion capture with multiple depth cameras. In: Computer and Robot Vision (CRV), 2016 13th Conference on, IEEE (2016) 24–31
11. Zhang, L., Sturm, J., Cremers, D., Lee, D.: Real-time human motion tracking using multiple depth cameras. In: Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on, IEEE (2012) 2389–2395
12. Phan, A., Ferrie, F.P.: Towards 3d human posture estimation using multiple kinects despite self-contacts. In: Machine Vision Applications (MVA), 2015 14th IAPR International Conference on, IEEE (2015) 567–571

13. Elhayek, A., de Aguiar, E., Jain, A., Thompson, J., Pishchulin, L., Andriluka, M., Bregler, C., Schiele, B., Theobalt, C.: Marconiconvnet-based marker-less motion capture in outdoor and indoor scenes. IEEE transactions on pattern analysis and machine intelligence **39** (2017) 501–514
14. Güler, R.A., Neverova, N., Kokkinos, I.: Densepose: Dense human pose estimation in the wild. arXiv preprint arXiv:1802.00434 (2018)
15. Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., Blake, A.: Real-time human pose recognition in parts from single depth images. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, Ieee (2011) 1297–1304
16. Girshick, R., Shotton, J., Kohli, P., Criminisi, A., Fitzgibbon, A.: Efficient regression of general-activity human poses from depth images. In: Computer Vision (ICCV), 2011 IEEE International Conference on, IEEE (2011) 415–422
17. Fang, H., Xie, S., Tai, Y.W., Lu, C.: Rmpe: Regional multi-person pose estimation. In: The IEEE International Conference on Computer Vision (ICCV). Volume 2. (2017)
18. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2014) 580–587
19. Song, L., Wu, W., Guo, J., Li, X.: Survey on camera calibration technique. In: Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2013 5th International Conference on. Volume 2., IEEE (2013) 389–392
20. Bradski, G., Kaehler, A.: Opencv. Dr. Dobbs journal of software tools (2000)
21. Felzenszwalb, P.F., Huttenlocher, D.P.: Pictorial structures for object recognition. International journal of computer vision **61** (2005) 55–79
22. Jiang, H.: Finding human poses in videos using concurrent matching and segmentation. In: Asian Conference on Computer Vision, Springer (2010) 228–243
23. Ganapathi, V., Plagemann, C., Koller, D., Thrun, S.: Real-time human pose tracking from range data. In: European conference on computer vision, Springer (2012) 738–751
24. Anguelov, D., Srinivasan, P., Koller, D., Thrun, S., Rodgers, J., Davis, J.: Scape: shape completion and animation of people. In: ACM transactions on graphics (TOG). Volume 24., ACM (2005) 408–416
25. Bogo, F., Kanazawa, A., Lassner, C., Gehler, P., Romero, J., Black, M.J.: Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In: European Conference on Computer Vision, Springer (2016) 561–578
26. Andriluka, M., Roth, S., Schiele, B.: Monocular 3d pose estimation and tracking by detection. In: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, IEEE (2010) 623–630
27. Sapp, B., Jordan, C., Taskar, B.: Adaptive pose priors for pictorial structures. In: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, IEEE (2010) 422–429
28. Zhu, Q., Yeh, M.C., Cheng, K.T., Avidan, S.: Fast human detection using a cascade of histograms of oriented gradients. In: Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on. Volume 2., IEEE (2006) 1491–1498
29. Lowe, D.G.: Object recognition from local scale-invariant features. In: Computer vision, 1999. The proceedings of the seventh IEEE international conference on. Volume 2., Ieee (1999) 1150–1157
30. Jain, A., Tompson, J., LeCun, Y., Bregler, C.: Modeep: A deep learning framework using motion features for human pose estimation. In: Asian conference on computer vision, Springer (2014) 302–315

31. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. (2012) 1097–1105
32. Insafutdinov, E., Andriluka, M., Pishchulin, L., Tang, S., Levinkov, E., Andres, B., Schiele, B.: Arttrack: Articulated multi-person tracking in the wild. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Volume 4327. (2017)
33. Girdhar, R., Gkioxari, G., Torresani, L., Paluri, M., Tran, D.: Detect-and-track: Efficient pose estimation in videos. arXiv preprint arXiv:1712.09184 (2017)
34. Johnson, S., Everingham, M.: Clustered pose and nonlinear appearance models for human pose estimation. In: Proceedings of the British Machine Vision Conference. (2010) doi:10.5244/C.24.12.
35. Eichner, M., Ferrari, V.: We are family: Joint pose estimation of multiple persons. In: European conference on computer vision, Springer (2010) 228–242
36. Ganapathi, V., Plagemann, C., Koller, D., Thrun, S.: Real time motion capture using a single time-of-flight camera. In: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, IEEE (2010) 755–762
37. Kazemi, V., Sullivan, J.: Using richer models for articulated pose estimation of footballers. In: BMVC. (2012)
38. Kazemi, V., Burenius, M., Azizpour, H., Sullivan, J.: Multi-view body part recognition with random forests. In: 2013 24th British Machine Vision Conference, BMVC 2013; Bristol; United Kingdom; 9 September 2013 through 13 September 2013, British Machine Vision Association (2013)
39. Sapp, B., Taskar, B.: Modec: Multimodal decomposable models for human pose estimation. In: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, IEEE (2013) 3674–3681
40. Sigal, L., Balan, A.O., Black, M.J.: Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. International journal of computer vision **87** (2010) 4
41. Ionescu, C., Papava, D., Olaru, V., Sminchisescu, C.: Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. IEEE transactions on pattern analysis and machine intelligence **36** (2014) 1325–1339
42. Andriluka, M., Iqbal, U., Milan, A., Insafutdinov, E., Pishchulin, L., Gall, J., Schiele, B.: Posetrack: A benchmark for human pose estimation and tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2018) 5167–5176
43. Moeslund, T.B., Hilton, A., Krüger, V.: A survey of advances in vision-based human motion capture and analysis. Computer vision and image understanding **104** (2006) 90–126
44. Yang, Y., Ramanan, D.: Articulated pose estimation with flexible mixtures-of-parts. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, IEEE (2011) 1385–1392
45. Jung, H.Y., Lee, S., Heo, Y.S., Yun, I.D., et al.: Random tree walk toward instantaneous 3d human pose estimation. In: CVPR. (2015) 2467–2474
46. Andriluka, M., Roth, S., Schiele, B.: Pictorial structures revisited: People detection and articulated pose estimation. In: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, IEEE (2009) 1014–1021
47. Tian, Y., Zitnick, C.L., Narasimhan, S.G.: Exploring the spatial hierarchy of mixture models for human pose estimation. In: European Conference on Computer Vision, Springer (2012) 256–269

48. Pishchulin, L., Andriluka, M., Gehler, P., Schiele, B.: Poselet conditioned pictorial structures. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2013) 588–595
49. Newell, A., Yang, K., Deng, J.: Stacked hourglass networks for human pose estimation. In: European Conference on Computer Vision, Springer (2016) 483–499
50. Wei, S.E., Ramakrishna, V., Kanade, T., Sheikh, Y.: Convolutional pose machines. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2016) 4724–4732
51. Chu, X., Yang, W., Ouyang, W., Ma, C., Yuille, A.L., Wang, X.: Multi-context attention for human pose estimation. arXiv preprint arXiv:1702.07432 **1** (2017)
52. Ning, G., Zhang, Z., He, Z.: Knowledge-guided deep fractal neural networks for human pose estimation. IEEE Transactions on Multimedia (2017)
53. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473 (2014)
54. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2016) 770–778
55. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A.: Inception-v4, inception-resnet and the impact of residual connections on learning. In: AAAI. Volume 4. (2017) 12
56. Pishchulin, L., Insafutdinov, E., Tang, S., Andres, B., Andriluka, M., Gehler, P.V., Schiele, B.: Deepcut: Joint subset partition and labeling for multi person pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2016) 4929–4937
57. Insafutdinov, E., Pishchulin, L., Andres, B., Andriluka, M., Schiele, B.: Deepercut: A deeper, stronger, and faster multi-person pose estimation model. In: European Conference on Computer Vision, Springer (2016) 34–50
58. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in neural information processing systems. (2015) 91–99
59. Newell, A., Huang, Z., Deng, J.: Associative embedding: End-to-end learning for joint detection and grouping. In: Advances in Neural Information Processing Systems. (2017) 2274–2284
60. Xia, F., Wang, P., Chen, X., Yuille, A.: Joint multi-person pose estimation and semantic part segmentation. arXiv preprint arXiv:1708.03383 (2017)
61. Ning, G., He, Z.: Dual path networks for multi-person human pose estimation. arXiv preprint arXiv:1710.10192 (2017)
62. Pfister, T., Charles, J., Zisserman, A.: Flowing convnets for human pose estimation in videos. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 1913–1921
63. Song, J., Wang, L., Van Gool, L., Hilliges, O.: Thin-slicing network: A deep structured model for pose estimation in videos. ArXiv170310898 Cs (2017)
64. Iqbal, U., Milan, A., Gall, J.: Posetrack: Joint multi-person pose estimation and tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Volume 1. (2017)
65. Jin, S., Ma, X., Han, Z., Wu, Y., Yang, W., Liu, W., Qian, C., Ouyang, W.: Towards multi-person pose tracking: Bottom-up and top-down methods. In: ICCV PoseTrack Workshop. (2017)
66. Daubney, B., Xie, X.: Tracking 3d human pose with large root node uncertainty. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, IEEE (2011) 1321–1328

67. Moon, S., Park, Y., Ko, D.W., Suh, I.H.: Multiple kinect sensor fusion for human skeleton tracking using kalman filtering. International Journal of Advanced Robotic Systems **13** (2016) 65