

Deep Dual Consecutive Network for Human Pose Estimation

Zhenguang Liu

Zhejiang Gongshang University

liuzhenguang2008@gmail.com

Haoming Chen

Zhejiang Gongshang University

chenhaomingbob@gmail.com

Runyang Feng*

Zhejiang Gongshang University

huaqinew2019@gmail.com

Shuang Wu

Nanyang Technological University

wushuang@outlook.sg

Shouling Ji*

Zhejiang University

sjj@zju.edu.cn

Bailin Yang*

Zhejiang Gongshang University

ybl@mail.zjgsu.edu.cn

Xun Wang*

Zhejiang Gongshang University

xwang@zjgsu.edu.cn

Abstract

*Multi-frame human pose estimation in complicated situations is challenging. Although state-of-the-art human joints detectors have demonstrated remarkable results for static images, their performances come short when we apply these models to video sequences. Prevalent shortcomings include the failure to handle motion blur, video defocus, or pose occlusions, arising from the inability in capturing the temporal dependency among video frames. On the other hand, directly employing conventional recurrent neural networks incurs empirical difficulties in modeling spatial contexts, especially for dealing with pose occlusions. In this paper, we propose a novel multi-frame human pose estimation framework, leveraging abundant temporal cues between video frames to facilitate keypoint detection. Three modular components are designed in our framework. A Pose Temporal Merger encodes keypoint spatiotemporal context to generate effective searching scopes while a Pose Residual Fusion module computes weighted pose residuals in dual directions. These are then processed via our Pose Correction Network for efficient refining of pose estimations. Our method ranks No.1 in the Multi-frame Person Pose Estimation Challenge on the large-scale benchmark datasets **PoseTrack2017** and **PoseTrack2018**. We have released our code, hoping to inspire future research.*

1. Introduction

Human pose estimation is a fundamental problem in computer vision, which aims at locating anatomical key-

points (e.g., wrist, ankle, etc.) or body parts. It has enormous applications in diverse domains such as security, violence detection, crowd riot scene identification, human behavior understanding, and action recognition [22]. Earlier methods [40, 38, 48, 29] adopt the probabilistic graphical model or the pictorial structure model. Recent methods have built upon the success of deep convolutional neural networks (CNNs) [5, 36, 41, 12, 27, 9, 25, 23, 24], achieving outstanding performance in this task. Unfortunately, most of the recent state-of-the-art methods are designed for static images, with greatly diminished performance when handling video input.

In this paper, we focus on the problem of multi-person pose estimation in video sequences. Conventional image-based approaches disregard the temporal dependency and geometric consistency across video frames. Dismissing these additional cues results in failure cases when dealing with challenging situations that inherently occurs in video sequences such as motion blur, video defocus, or pose occlusions. Effectively leveraging the temporal information in video sequences is of great significance to facilitate pose estimation and often plays an indispensable role for detecting heavily occluded or blurry joints.

A direct and intuitive approach to tackle this issue is to employ recurrent neural networks (RNNs) such as Long-Short Term Memory (LSTM), Gate Recurrent Unit (GRU) or 3DCNNs to model geometric consistency as well as temporal dependency across video frames. [25] uses convolutional LSTM to capture temporal and spatial cues, and directly predicts the keypoint heatmap sequences for videos. This RNN based approach is more effective when the human subjects are spatially sparse such as single-person scenes with minimal occlusion. However, performance

*Corresponding Authors



Figure 1. An illustration of our Pose Temporal Merger (PTM) network. **(a)**: Original video sequence in the datasets, and we aim to detect poses in the current frame F_c . **(b)**: Each person in the original video sequence is assembled to a cropped clip and a single-person joints detector gives preliminary estimations of keypoint heatmaps (illustrated for right wrist). **(c-left)**: Merged keypoint heatmaps for the right wrist, generated by our PTM network through encoding the keypoint spatial contexts. The color intensity encodes spatial aggregation. **(c-right)**: Zoomed-in view of the merged keypoint heatmaps.

is severely hindered in the case of occlusion commonly occurring in multi-person pose estimation and even self-occlusion in the single-person case. [39] proposes a 3DHR-Net (extension of HRNet [33] to include a temporal dimension) for extracting spatial and temporal features across video frames to estimate pose sequences. This model has shown excellent results particularly for adequately long duration single-person sequences. Another line of work considers fine-tuning the primary prediction with high confidence keypoints from the adjacent frames. [31, 28] propose to compute the dense optical flow between every two frames, and leverage the additional flow based representations to align the predictions. This approach is promising when the optical flow can be computed precisely. However in cases involving motion blur or defocus, the poor image qualities lead to imprecise optical flows which translates to performance drops.

To address the shortcomings of existing methods, we propose to incorporate consecutive frames from dual temporal directions to improve pose estimation in videos. Our framework, termed Dual Consecutive network for pose estimation (DCPose), first encodes the spatial-temporal key-

point context into localized search scopes, computes pose residuals, and subsequently refines the keypoint heatmap estimations. Specifically, we design three task-specific modules within the DCPose pipeline. 1) As illustrated in Fig. 1, a Pose Temporal Merger (PTM) network performs keypoints aggregation over a continuous video segment (e.g., three consecutive frames) with group convolution, thereby localizing the search range for the keypoint. 2) A Pose Residual Fusion (PRF) network is introduced to efficiently obtain the pose residuals between the current frame and adjacent frames. PRF computes inter-frame keypoint offsets by explicitly utilizing the temporal distance. 3) Finally, a Pose Correction Network (PCN) comprising five parallel convolution layers with different dilation rates is proposed for resampling keypoint heatmaps in the localized search range.

It is worth mentioning that the architecture of our network extends the successful PoseWarper architecture [3] in three ways. (1) PoseWarper focuses on enabling effective label propagation between frames, while we aim to refine the pose estimation of current frame using the motion context and temporal information from unlabeled neighboring frames. (2) Information from two directions are utilized and we explicitly consider weighted residuals between frames. (3) Instead of applying the learned warping operation to a heatmap from one adjacent frame, the new network fuses together heatmaps from the adjacent frames and the current frame.

To summarize, our key contributions are: 1) A novel dual consecutive pose estimation framework is proposed. DCPose effectively incorporates bidirectional temporal cues across frames to facilitate the multi-person pose estimation task in videos. 2) We design 3 modular networks within DCPose to effectively utilise the temporal context: i) a novel Pose Temporal Merger network for effectively aggregating keypoint across frames and identifying a search scope, ii) a Pose Residual Fusion network to efficiently compute weighted pose residuals across frames, and iii) a Pose Correction Network that updates the pose estimation with the refined search scope and pose residual information. 3) Our method achieves state-of-the-art results on PoseTrack2017 and PoseTrack2018 Multi-frame Person Pose Estimation Challenge. To facilitate future research, our source code is released at <https://github.com/Pose-Group/DCPose>.

2. Related Work

2.1. Imaged Based Multi Person Pose Estimation

Earlier image-based human pose estimation works generally fall within a pictorial structure model paradigm, in which the human body is represented as a tree-structured model [40, 38, 48, 29] or a forest model [34, 10]. Despite

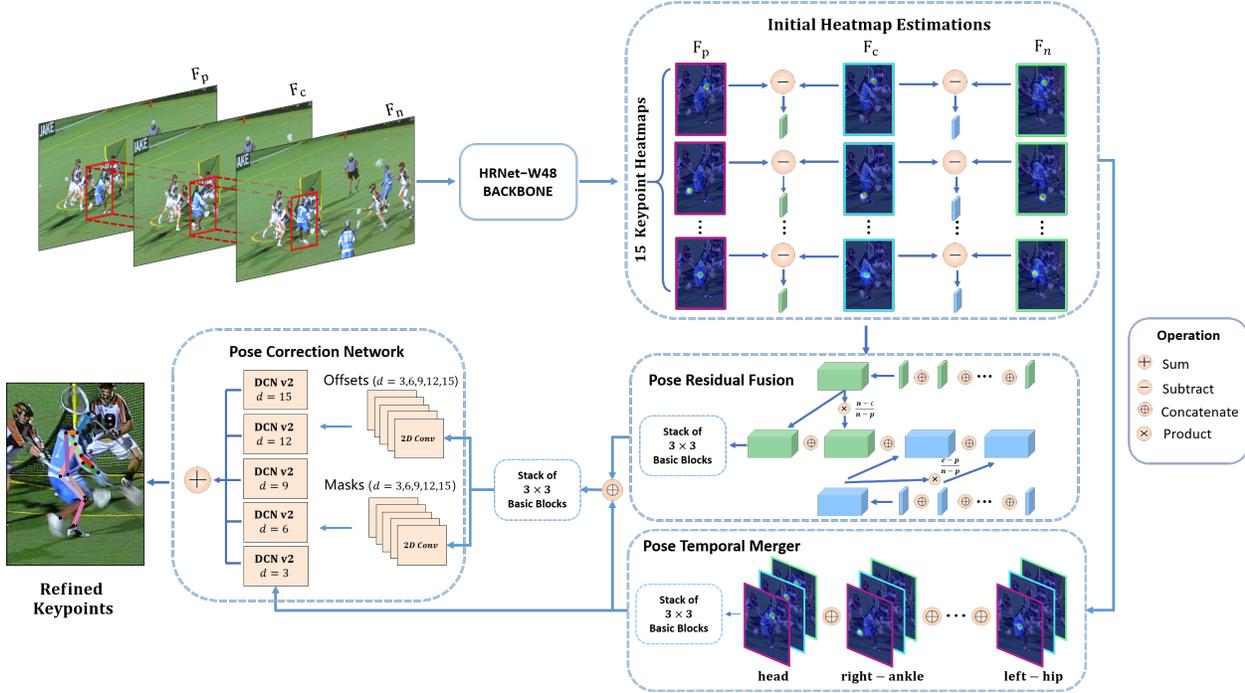


Figure 2. Overall pipeline of our DCPose framework. The goal is to locate the keypoint positions for the current frame F_c . First, an individual person i is assembled into an input sequence $\text{Clip}_i(p, c, n)$, and a HRNet backbone predicts initial keypoint heatmaps $\mathbf{h}_i(p), \mathbf{h}_i(c), \mathbf{h}_i(n)$. Our Pose Temporal Merger (PTM) and Pose Residual Fusion (PRF) networks work concurrently to obtain an effective search scope $\Phi_i(p, c, n)$ and pose residuals $\Psi_i(p, c, n)$, respectively. These are then fed into our Pose Correction Network (PCN) which refines the keypoint estimation for person i in F_c .

allowing for efficient inference, these approaches tend to be insufficient in modeling complex relationships between body parts, and this weakness is accentuated when temporal information enters the picture. Recently, neural networks based methods [20, 21, 35, 2, 7, 14, 46, 37, 32, 8, 50] have been in the spotlight due to their superior performance in various fields. One line of work [6] outputs skeletal joints coordinates directly by regressing image features. Another approach [27, 28] utilises probability heatmaps to represent joints locations. Due to the reduced difficulty for optimization, heatmap based pose estimation have since been widely adopted. In general, these methods can be classified into part-based framework (bottom-up) and two-step framework (top-down). The bottom-up approach [5, 17, 18, 19] first detects individual body parts, then assembles these constituent parts into the entire person. [5] builds a bottom-up pipeline and utilizes part affinity fields to capture pairwise relationships between different body parts. Conversely, the top-down approach [12, 43, 41, 33, 27, 26] first performs person detection, then proceeds with single-person pose estimation on each individual. [41] proposes a sequential architecture of convolutional pose machines, which follows a strategy of iteratively refining the output of each network stage. [12] designs a symmetric spatial transformer network

for extracting a high-quality single person region from an inaccurate bounding box. A recent work in [33] proposes a HRNet that performs multi-scale fusion to retain high resolution feature maps. This improves spatial precision in keypoint heatmaps and achieves the state-of-the-art on several image-based benchmarks.

2.2. Video Based Multi Person Pose Estimation

Directly applying the existing image-level methods to video sequences produces unsatisfactory predictions, primarily due to the failure to capture temporal dependency among video frames. Consequently, these models fail to handle motion blur, video defocus, or pose occlusions which are frequently encountered in video inputs. [28, 31, 42] compute the dense optical flow between every consecutive frames with the flow representations providing additional cues for aligning predictions. However, motion blur, defocus or occlusion occurrences hinder optical flow computation and affect performance. [25] replaces the convolutional pose machines in [41] with convolutional LSTMs for modeling temporal information in addition to spatial contexts. A principal shortcoming of such an approach is being severely impacted by occlusion. [3] proposes to learn an effective video pose detector from sparsely labeled videos

through a warping mechanism and has turned out to be very successful, dominating the PoseTrack leaderboard for a long time. [39] extends HRNet [33] with temporal convolutions and proposes 3DHRNet, which is successful in handling pose estimation and tracking jointly.

3. Our Approach

The pipeline of our proposed DCPose is illustrated in Fig. 2. To improve keypoint detection for the current frame F_c , we make use of additional temporal information from a previous frame F_p and a future frame F_n . F_p and F_n are selected within a frame window $[c - T, c + T]$, where $p \in [c - T, c)$ and $n \in (c, c + T]$ respectively denote the frame indices. The bounding boxes for individual persons in F_c are first obtained by a human detector. Each bounding box is enlarged by 25% and is further used to crop the same person in F_p and F_n . Individual i in the video will thus be composed of a cropped video segment, which we denote as $\text{Clip}_i(p, c, n)$. $\text{Clip}_i(p, c, n)$ is then fed into a backbone network that serves to output preliminary keypoint heatmap estimations $\mathbf{h}_i(p, c, n)$. The pose heatmaps $\mathbf{h}_i(p, c, n)$ is then processed in parallel through two modular networks, Pose Temporal Merger (PTM) and Pose Residual Fusion (PRF). PTM outputs $\Phi_i(p, c, n)$, which encodes the spatial aggregation, and PRF computes $\Psi_i(p, c, n)$, which captures pose residuals in two directions. Both feature tensors $\Phi_i(p, c, n)$ and $\Psi_i(p, c, n)$ are then simultaneously fed into our Pose Correction Network (PCN) to refine and improve upon the initial pose estimations. In what follows, we introduce the three key components in detail.

3.1. Pose Temporal Merger

The motivation for our Pose Temporal Merger (PTM) comes from the following observations and heuristics. 1) Although existing pose estimation methods such as [33, 12] suffer from performance deterioration on videos, we observe that their predictions do still provide useful information for approximating the keypoint spatial positions. 2) Another heuristic is on temporal consistency, *i.e.*, the pose of an individual does not undergo dramatic and abrupt changes across a very few frame intervals (typically 1/60 to 1/25 of a second per frame). Therefore, we design PTM to encode the keypoint spatial contexts based on initial predictions (from a backbone network), providing a compressed search scope that facilitates refinement and correction of pose prediction within a confined range.

For person i , the backbone network returns initial keypoint heatmaps $\mathbf{h}_i(p), \mathbf{h}_i(c), \mathbf{h}_i(n)$. Naively, we could merge them through a direct summation $\mathbf{H}_i(p, c, n) = \mathbf{h}_i(p) + \mathbf{h}_i(c) + \mathbf{h}_i(n)$. However, we expect that the additional information that may be extracted from F_p and F_n is inversely proportional to their temporal distances from

current frame F_c . We formalize this intuition as:

$$\mathbf{H}(p, c, n) = \frac{n - c}{n - p} \mathbf{h}_i(p) + \mathbf{h}_i(c) + \frac{c - p}{n - p} \mathbf{h}_i(n). \quad (1)$$

Recall that p, c, n are frame indices. We explicitly assign higher weights to the frames that are temporally nearer to the current frame.

Based on the important fact that convolution operations serve to adjust (feature) weights, we utilise convolutional neural networks to practically implement the idea of Eq. 1. However, including all joint channels in the computation for the merged keypoint heatmap of a single joint will result in redundancy. For example, when encoding the spatial context of the left wrist, involving other joints such as the head and ankle at different times will likely not to have any bearing and may even breed confusion. Therefore, for each joint, we only include its own specific temporal information for computing its merged keypoint heatmap. This is implemented via a group convolution. We regroup the keypoint heatmaps $\mathbf{h}_i(p), \mathbf{h}_i(c), \mathbf{h}_i(n)$ according to joint, and stack them to a feature tensor ϕ_i , which can be expressed as:

$$\phi_i(p, c, n) = \bigoplus_{j=1}^N \frac{n - c}{n - p} \mathbf{h}_i^j(p) \oplus \mathbf{h}_i^j(c) \oplus \frac{c - p}{n - p} \mathbf{h}_i^j(n) \quad (2)$$

where \oplus denotes the concatenate operation and the superscript j index the j -th joint for a total of N joints. Subsequently, the feature tensor ϕ_i is fed into a stack of 3×3 residual blocks (adapted from the residual steps block in RSN [4]), producing the merged keypoint heatmaps $\Phi_i(p, c, n)$:

$$\phi_i(p, c, n) \xrightarrow[\text{residual blocks}]{\text{stack of } 3 \times 3} \Phi_i(p, c, n). \quad (3)$$

This group convolution not only eliminates the disturbance of irrelevant joints, but also removes redundancy and shrinks the amount of model parameters required. It is also advantageous to directly summing keypoint heatmaps in Eq. 1 since the group CNN operation allow different weights at the pixel level and benefits learning an end-to-end model. Visual results of aggregated keypoint heatmaps following our PTM is illustrated in Fig. 1.

3.2. Pose Residual Fusion

Parallel to spatial aggregation of keypoint heatmaps in PTM, our Pose Residual Fusion (PRF) branch aims to compute the pose residuals which will serve as additional favorable temporal cues. Given keypoint heatmaps $\mathbf{h}_i(p), \mathbf{h}_i(c), \mathbf{h}_i(n)$, we compute the pose residual features

as follows:

$$\begin{aligned}
 \psi_i(p, c) &= \mathbf{h}_i(c) - \mathbf{h}_i(p) \\
 \psi_i(c, n) &= \mathbf{h}_i(n) - \mathbf{h}_i(c) \\
 \psi_i &= \psi_i(p, c) \oplus \psi_i(c, n) \\
 &\oplus \frac{n-c}{n-p} \psi_i(p, c) \oplus \frac{c-p}{n-p} \psi_i(c, n).
 \end{aligned} \tag{4}$$

ψ_i concatenates the original pose residuals $\psi_i(p, c)$, $\psi_i(c, n)$, and their weighted versions, where the weights are obtained according to the temporal distances. Similar to PTM, ψ_i is then processed via a stack of 3×3 residual blocks to give the final pose residual feature $\Psi_i(p, c, n)$:

$$\psi_i(p, c, n) \xrightarrow[\text{residual blocks}]{\text{stack of } 3 \times 3} \Psi_i(p, c, n). \tag{5}$$

3.3. Pose Correction Network

Given the merged keypoint heatmaps $\Phi_i(p, c, n)$ and pose residual feature tensor $\Psi_i(p, c, n)$, our Pose Correction Network is employed to refine the initial keypoint heatmap estimation $\mathbf{h}_i(c)$, yielding adjusted final keypoint heatmaps. Primarily, the pose residual feature tensor $\Psi_i(p, c, n)$ is used as the input for five parallel 3×3 convolution layers with different dilation rates $d \in \{3, 6, 9, 12, 15\}$. This computation gives five groups of offsets for the five kernels of the subsequent deformable convolution layer. Formally, the offsets are computed as:

$$\Phi_i(p, c, n) \oplus \Psi_i(p, c, n) \xrightarrow[\text{residual blocks}]{\text{stack of } 3 \times 3} \xrightarrow[\text{convolution layers}]{\text{dilation rate } d} O_{i,d}. \tag{6}$$

Different dilation rates correspond to varying the size of the effective receptive field whereby enlarging the dilation rate [45] increases the scope of the receptive field. A smaller dilation rate focuses on local appearance, which is more sensitive for capturing subtle motion contexts. Conversely, using a large dilation rate allows us to encode global representations and capture relevant information of a larger spatial scope. In addition to the offset computation, we feed the merged keypoint heatmaps to similar convolution layers and obtain five sets of masks M_d as:

$$\Phi_i(p, c, n) \oplus \Psi_i(p, c, n) \xrightarrow[\text{residual blocks}]{\text{stack of } 3 \times 3} \xrightarrow[\text{convolution layers}]{\text{dilation rate } d} M_{i,d}. \tag{7}$$

The parameters of two dilation convolution structures for offset O and mask M computation are independent. A mask M_d can be considered as the weight matrix for a convolution kernel.

We implement the pose correction module through the deformable convolution V2 network (DCN v2 [49]) at various dilation rates d . DCN v2 takes the following inputs: 1) the merged keypoint heatmaps $\Phi_i(p, c, n)$, 2) the kernel offsets $O_{i,d}$, and 3) the masks $M_{i,d}$, and outputs a pose

Method	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	Mean
PoseFlow[44]	66.7	73.3	68.3	61.1	67.5	67.0	61.3	66.5
JointFlow[11]	-	-	-	-	-	-	-	69.3
FastPose[47]	80.0	80.3	69.5	59.1	71.4	67.5	59.4	70.3
SimpleBaseline[43]	81.7	83.4	80.0	72.4	75.3	74.8	67.1	76.7
STEmbedding[16]	83.8	81.6	77.1	70.0	77.4	74.5	70.8	77.0
HRNet[33]	82.1	83.6	80.4	73.3	75.5	75.3	68.5	77.3
MDPN[13]	85.2	88.5	83.9	77.5	79.0	77.0	71.4	80.7
PoseWarper[3]	81.4	88.3	83.9	78.0	82.4	80.5	73.6	81.2
DCPose	88.0	88.7	84.1	78.4	83.0	81.4	74.2	82.8

Table 1. Quantitative Results (AP) on PoseTrack2017 validation set.

Method	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	Total
PoseFlow[44]	64.9	67.5	65.0	59.0	62.5	62.8	57.9	63.0
JointFlow[11]	-	-	-	53.1	-	-	50.4	63.4
KeyTrack[30]	-	-	-	71.9	-	-	65.0	74.0
DetTrack[39]	-	-	-	69.8	-	-	65.9	74.1
SimpleBaseline[43]	80.1	80.2	76.9	71.5	72.5	72.4	65.7	74.6
HRNet[33]	80.1	80.2	76.9	72.0	73.4	72.5	67.0	74.9
PoseWarper[3]	79.5	84.3	80.1	75.8	77.6	76.8	70.8	77.9
DCPose	84.3	84.9	80.5	76.1	77.9	77.1	71.2	79.2

Table 2. Performance comparisons on the PoseTrack2017 test set.

Method	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	Mean
AlphaPose[12]	63.9	78.7	77.4	71.0	73.7	73.0	69.7	71.9
MDPN[13]	75.4	81.2	79.0	74.1	72.4	73.0	69.9	75.0
PoseWarper[3]	79.9	86.3	82.4	77.5	79.8	78.8	73.2	79.7
DCPose	84.0	86.6	82.7	78.0	80.4	79.3	73.8	80.9

Table 3. Quantitative Results(AP) on PoseTrack2018 validation set.

heatmap for person i at dilation rate d :

$$(\Phi_i(p, c, n), O_{i,d}, M_{i,d}) \xrightarrow[\text{DCN v2}]{\text{dilation rate } d} \mathbf{H}_{i,d}(c). \tag{8}$$

The five outputs for five dilation rates are summarized and normalized to yield the final pose prediction for person i :

$$\sum_{d \in \{3, 6, 9, 12, 15\}} \mathbf{H}_{i,d}(c) \xrightarrow{\text{normalization}} \mathbf{H}_i(c). \tag{9}$$

Ultimately, the above procedure is performed for each individual i . By effectively utilising the additional cues from F_p and F_n in our DCPose framework, the final pose heatmaps are enhanced and improved.

3.4. Implementation Details

Backbone Model Our network is highly adaptable and we can seamlessly integrate any image based pose estimation architecture as our backbone. We employ the state-of-the-art Deep High Resolution Network (HRNet-W48 [33]) as our backbone joints detector, since its superior performance for single image pose estimation will be beneficial for our approach.

Training Our Deep Dual Consecutive Network is implemented in PyTorch. During training, we use the ground truth person bounding boxes to generate the $\mathbf{Clip}_i(p, c, n)$ for person i as the input sequence to our model. For boundary cases, we apply same padding. In other words, if there

Method	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	Total
AlphaPose++ ^[12, 13]	-	-	-	66.2	-	-	65.0	67.6
DetTrack ^[39]	-	-	-	69.8	-	-	67.1	73.5
MDPN ^[13]	-	-	-	74.5	-	-	69.0	76.4
PoseWarper ^[3]	78.9	84.4	80.9	76.8	75.6	77.5	71.8	78.0
DCPose	82.8	84.0	80.8	77.2	76.1	77.6	72.3	79.0

Table 4. Performance comparisons on the PoseTrack2018 test set.

are no frames to extend forward and backward from F_c , F_p or F_n will be replaced by F_c . We utilize the HRNet-W48 pretrained on the PoseTrack dataset as our backbone, and freeze the backbone parameters throughout training, only backpropagating through the subsequent components in DCPose.

Loss function We employ the standard pose estimation loss function as our cost function. Training aims to minimize the total Euclidean or L_2 distance between prediction and ground truth heatmaps for all joints. The cost function is defined as:

$$L = \frac{1}{N} * \sum_{j=1}^N v_j * ||G(j) - P(j)||^2 \quad (10)$$

Where $G(j)$, $P(j)$ and v_j respectively denote the ground truth heatmap, prediction heatmap and visibility for joint j . During training, the total number of joints is set to $N = 15$. The ground truth heatmaps are generated via a 2D Gaussian centered at the joint location.

4. Experiments

In this section, we present our experimental results on two large-scale benchmark datasets: Posetrack2017 and PoseTrack2018 Multi-frame Person Pose Estimation Challenge datasets.

4.1. Experimental Settings

Datasets PoseTrack is a large-scale public dataset for human pose estimation and articulated tracking in video and includes challenging situations with complicated movement of highly occluded people in crowded environments. The PoseTrack2017 dataset [15] contains 514 video clips and 16,219 pose annotations, with 250 clips for training, 50 clips for validation, and 214 clips for testing. The PoseTrack2018 dataset greatly increased the number of video clips to a total of 1,138 clips and 153,615 pose annotations. The training, validation, and testing splits consist of 593, 170, and 375 clips respectively. The 30 frames in the center of the training video clips are densely annotated. For the validation clips, annotations are provided every four frames. Both PoseTrack2017 and PoseTrack2018 identify 15 joints, with an additional annotation label for joint visibility. We evaluate our model only for visible joints with the average precision (AP) metric [15, 1].

Parameter Settings During training, we incorporate data augmentation including random rotations, scaling, truncating, and horizontal flipping to increase variation. Input image size is fixed to 384×288 . The default interval between F_c and F_p or F_n is set to 1. Backbone parameters are fixed to the pretrained HRNet-W48 model weights. All subsequent weight parameters are initialized from a Gaussian distribution with $\mu = 0$ and $\sigma = 0.001$, while bias parameters are initialized to 0. We employ the Adam optimizer with a base learning rate of 0.0001 that decays by 10% every 4 epochs. We train our model for a batch size of 32 for 20 epochs with 2 Nvidia GeForce Titan X GPUs.

4.2. Comparison with State-of-the-art Approaches

Results on the PoseTrack2017 Dataset We evaluate our approach on PoseTrack2017 validation set and full test set using the widely adopted average precision (AP) metric [43, 44, 11, 47]. Table 1 presents the quantitative results of different approaches in terms of AP on PoseTrack2017 validation set. We benchmark our DCPose model against eight existing methods [44], [11], [47], [43], [16], [33], [13], [3]. In Table 1, the APs of key joints, such as Head, Shoulder, Knee, and Elbow, are reported, as well as the mAP (mean AP) for all joints.

Results on the test set are provided in Table 2. These results are obtained by uploading our prediction results to the PoseTrack evaluation server: <https://posetrack.net/leaderboard.php> because the annotations for test set are not public. Our DCPose network achieves state-of-the-art results for multi-frame person pose estimation challenge for both the validation and test sets. DCPose consistently outperforms existing methods and achieves a **mAP** of 79.2. The performance boost for relatively difficult joints is also encouraging: we obtain an **mAP** of 76.1 for the wrist and an **mAP** of 71.2 for the ankle. Some sample results are displayed in Fig. 3, which are indicative of the effectiveness of our method in complex scenes. More visualized results can be found in <https://github.com/Pose-Group/DCPose>.

Results on the PoseTrack2018 Dataset We also evaluate our model on the PoseTrack2018 dataset. The validation and test set AP results are tabulated in Table 3 and Table 4, respectively. As shown in the tables, our approach once again delivers the state-of-the-art results. We achieve an **mAP** of 79.0 on the test set, and obtain an **mAP** of 77.2 for the difficult *wrist* joint and 72.3 for the *ankle*.

4.3. Ablation Experiments

Extensive ablation experiments are performed on the PoseTrack2017 dataset to study the effectiveness of various components in our DCPose framework. Through ablating the various modular networks including Pose Temporal Merger (PTM), Pose Residual Fusion (PRF) and Pose Cor-

Method	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	Mean
DCPose, complete, $T = 1$	88.0	88.7	84.1	78.4	83.0	81.4	74.2	82.8
r/m PTM, $\Phi(p, c, n) \leftarrow \mathbf{h}(p) + \mathbf{h}(c) + \mathbf{h}(n)$	87.3	88.4	83.6	77.6	82.8	78.4	73.7	82.0
r/m PTM, $\Phi(p, c, n) \leftarrow \mathbf{h}(c)$	87.5	88.5	83.7	78.1	82.9	80.8	73.4	82.2
r/m PRF	87.4	88.3	83.6	77.6	82.7	80.5	73.4	82.1
1 dilation convolution, $d = 3$	87.3	88.5	83.8	72.6	82.9	78.4	73.7	81.7
2 dilation convolutions, $d \in \{3, 6\}$	87.5	88.2	83.6	74.7	82.8	79.8	73.6	81.9
3 dilation convolutions, $d \in \{3, 6, 9\}$	87.9	88.6	83.9	78.3	82.9	81.4	74.0	82.7
4 dilation convolutions, $d \in \{3, 6, 9, 12\}$	87.9	88.6	84.0	76.9	83.0	80.9	74.0	82.6
r/m previous frame	87.8	88.4	83.8	77.8	82.6	80.6	73.7	82.2
r/m next frame	88.0	88.6	83.7	77.7	82.8	80.7	73.2	82.3
$T = 2$	87.7	88.4	83.8	78.0	83.0	80.8	73.2	82.6
$T = 3$	87.6	88.4	83.6	77.7	82.6	80.7	71.7	82.1
$T = 4$	81.2	87.4	83.3	76.7	81.9	80.2	72.1	81.7

Table 5. Ablation study of different components in our DCPose performed on PoseTrack2017 validation set. “r/m X” refers to removing X module in our network. The complete DCPose consistently achieves the best results which are highlighted.



Figure 3. Visual results of our model on the PoseTrack2017 and PoseTrack2018 datasets comprising complex scenes: high-speed movement, occlusion, and multiple persons.

rection Network, we evaluate their contributions towards the overall performance. We also investigate the efficacy of including information from both temporal directions as well as the impact of modifying the temporal distance, *i.e.*, frame intervals between F_c and F_p, F_n . The results are reported in Table 5.

Pose Temporal Merger For this ablation setting, we remove the PTM and instead obtain the merged pose heatmaps $\Phi(p, c, n)$ as follows: i) $\mathbf{h}(p) + \mathbf{h}(c) + \mathbf{h}(n) \xrightarrow[convolution\ layer]{3 \times 3} \Phi(p, c, n)$; ii) $\mathbf{h}(c) \rightarrow \Phi(p, c, n)$. The **mAP** falls from 82.8 to 82.0 for (i) and 82.2 for (ii). This significant performance degradation upon removal of the PTM module can be attributed to the failure of obtaining an effective search range for the joints, which lead to decreased accuracy in locating the joint in the subsequent pose correction stage.

Pose Residual Fusion We investigate removing the PRF and compute the pose residual maps $\Psi(p, c, n)$ with the following scheme: $\mathbf{h}(c) - \mathbf{h}(p) \oplus \mathbf{h}(n) - \mathbf{h}(c) \xrightarrow[convolution\ layer]{3 \times 3} \Psi(p, c, n)$. This results in the **mAP** dropping 0.7 to 82.1. This significant reduction in performance highlights the important role of PRF in providing accurate pose residual cues for computing offsets and guiding the keypoint localization.

Pose Correction Network We study the effects of adopting different sets of dilation rates for the convolutions in PCN. This corresponds to different effective receptive fields. We experiment with four different dilation settings: $d = 3, d \in \{3, 6\}, d \in \{3, 6, 9\}$ and $d \in \{3, 6, 9, 12\}$ whereas the complete DCPose framework setting has $d \in \{3, 6, 9, 12, 15\}$. From the results in Table 5, we observe the gradual improvement of the **mAP** with increasing levels of



Figure 4. Visualization of the pose predictions of our DCPose model(a), SimpleBaseline(b), HRNet(c), and PoseWarper(d) on the challenging cases from the *PoseTrack2017* and *PoseTrack2018* datasets. Each column from left to right represents the Rapid-Motion, Nearby-Person, Occlusions, and Video-Defocus scene, respectively. Inaccurate predictions are highlighted with the red solid circles.

dilation rates, from $81.7 \rightarrow 81.9 \rightarrow 82.7 \rightarrow 82.6 \rightarrow 82.8$. This is in line with our intuitions that increasing the depth and scope of the effective receptive fields, *i.e.*, by varying the range of dilation rates, the Pose Correction Network is able to model local and global contexts more efficiently, leading to more accurate estimations of the joint locations.

Bidirectional Temporal Input In addition, we examine the effects of removing a single temporal direction, *i.e.*, removal of either F_p or F_n from the input to PTM and PRF. In removing the previous (respectively next) frame F_p (respectively F_n), the **mAP** drops 0.6 (respectively 0.5). This highlights the importance of leveraging dual temporal directions as each direction allows access to useful information that are beneficial in improving pose estimation for videos.

Time Interval T The time interval T described in Section 3 is a hyper-parameter whose default value is set to 1. In other words, DCPose looks at 3 consecutive frames as the shorter time/frame interval ensures the validity of our assumptions of temporal consistency. We experiment with enlarging the interval between frames, where T is set to 2, 3, 4. Indices p and n are randomly selected with $p \in [c-T, c)$ and $n \in (c, c+T]$. The results reflect a performance drop with an increase in T , whereby the **mAP** decreases from 82.8 for $T = 1$ to 82.6, 82.1, 81.7 as T goes to 2, 3, 4 respectively. This is in accordance with our expectations, since frame-wise difference is small for shorter time intervals so that F_p and F_n can provide more abundant and effective temporal and spatial cues for our PTM and PRF. Conversely, a large time interval would invalidate our assumption of time consistency as the frame differences would be too great and the additional cues would lack efficacy. This ablation experiment explains our choice of modeling con-

secutive frames. The continuous frames of a video clip facilitate a precise aggregation of spatiotemporal context cues and improve the robustness of our model.

4.4. Comparison of Visual Results

In order to evaluate the capability of our model to adapt to sophisticated scenarios, we illustrate in Fig. 4 the side-by-side comparisons of our DCPose network against state-of-the-art approaches. Each column depicts different scenario complications including *rapid motion*, *nearby-person*, *occlusions* and *video defocus*, whereas each row displays the joint detection results from different methods. We compare our a) DCPose against 3 state-of-the-art methods, namely b) SimpleBaseline [43], c) HRNet-W48 [33] and d) PoseWarper [3]. It is observed that our method yields more robust detection for such challenging cases. SimpleBaseline and HRNet-W48 are trained on static images and fail to capture temporal dependency among video frames, resulting in suboptimal joint detection. On the other hand, PoseWarper leverages the temporal information between video frames to warp the initial pose heatmaps but only employing one adjacent video frame as the auxiliary frame. Our DCPose network makes full use of the temporal context by processing consecutive frames in dual directions. Through our principled design of PTM and PRF modules for better encoding of these information as well as PCN for refining pose detection, our method achieves new state-of-the-art both quantitatively and qualitatively.

5. Conclusion

In this paper, we propose a dual consecutive network for multi-frame person pose estimation which significantly outperforms existing state-of-the-art methods on the benchmark datasets. We design a Pose Temporal Merger and a Pose Residual Fusion module that allows abundant auxiliary information to be drawn from the adjacent frames, providing a localized and pose residual corrected search range for location keypoints. Our Pose Correction Network employs multiple effective receptive fields to refine pose estimation in this search range, achieving notable improvements and is able to handle complex scenes.

6. Acknowledgements

This work was supported by the Natural Science Foundation of Zhejiang Province, China No. LQ19F020001, the National Natural Science Foundation of China No. 61902348, the Humanities and Social Science Fund of the Ministry of Education of China under Grant No.17YJCZH076, and the Zhejiang Science and Technology Project under Grant No.LGF18F020001.

References

- [1] Mykhaylo Andriluka, Umar Iqbal, Eldar Insafutdinov, Leonid Pishchulin, Anton Milan, Juergen Gall, and Bernt Schiele. Posetrack: A benchmark for human pose estimation and tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 6
- [2] Bruno Artacho and Andreas Savakis. Unipose: Unified human pose estimation in single images and videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7035–7044, 2020. 3
- [3] Gedas Bertasius, Christoph Feichtenhofer, Du Tran, Jianbo Shi, and Lorenzo Torresani. Learning temporal pose estimation from sparsely-labeled videos. In *Advances in Neural Information Processing Systems*, pages 3027–3038, 2019. 2, 3, 5, 6, 8
- [4] Yuanhao Cai, Zhicheng Wang, Zhengxiong Luo, Binyi Yin, Angang Du, Haoqian Wang, Xinyu Zhou, Erjin Zhou, Xiangyu Zhang, and Jian Sun. Learning delicate local representations for multi-person pose estimation. *arXiv preprint arXiv:2003.04030*, 2020. 4
- [5] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 1, 3
- [6] Joao Carreira, Pulkit Agrawal, Katerina Fragkiadaki, and Jitendra Malik. Human pose estimation with iterative error feedback. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4733–4742, 2016. 3
- [7] 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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5386–5395, 2020. 3
- [8] Zhiyong Cheng, Ying Ding, Xiangnan He, Lei Zhu, Xuemeng Song, and Mohan S. Kankanhalli. A³ncf: An adaptive aspect attention model for rating prediction. In *IJCAI*, pages 3748–3754, 2018. 3
- [9] Xiao Chu, Wei Yang, Wanli Ouyang, Cheng Ma, Alan L Yuille, and Xiaogang Wang. Multi-context attention for human pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1831–1840, 2017. 1
- [10] Matthias Dantone, Juergen Gall, Christian Leistner, and Luc Van Gool. Human pose estimation using body parts dependent joint regressors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3041–3048, 2013. 2
- [11] Andreas Doering, Umar Iqbal, and Juergen Gall. Joint flow: Temporal flow fields for multi person tracking. *arXiv preprint arXiv:1805.04596*, 2018. 5, 6
- [12] Hao-Shu Fang, Shuqin Xie, Yu-Wing Tai, and Cewu Lu. Rmpe: Regional multi-person pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 1, 3, 4, 5, 6
- [13] Hengkai Guo, Tang Tang, Guozhong Luo, Riwei Chen, Yongchen Lu, and Linfu Wen. Multi-domain pose network for multi-person pose estimation and tracking. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0, 2018. 5, 6
- [14] Junjie Huang, Zheng Zhu, Feng Guo, and Guan Huang. The devil is in the details: Delving into unbiased data processing for human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5700–5709, 2020. 3
- [15] Umar Iqbal, Anton Milan, and Juergen Gall. Posetrack: Joint multi-person pose estimation and tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 6
- [16] Sheng Jin, Wentao Liu, Wanli Ouyang, and Chen Qian. Multi-person articulated tracking with spatial and temporal embeddings. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5664–5673, 2019. 5, 6
- [17] Muhammed Kocabas, Salih Karagoz, and Emre Akbas. Multiposenet: Fast multi-person pose estimation using pose residual network. In *Proceedings of the European conference on computer vision (ECCV)*, pages 417–433, 2018. 3
- [18] Sven Kreiss, Lorenzo Bertoni, and Alexandre Alahi. Pifpaf: Composite fields for human pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11977–11986, 2019. 3
- [19] Jiefeng Li, Can Wang, Hao Zhu, Yihuan Mao, Hao-Shu Fang, and Cewu Lu. Crowdpose: Efficient crowded scenes pose estimation and a new benchmark. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10863–10872, 2019. 3
- [20] Kyaw Zaw Lin, Weipeng Xu, Qianru Sun, Christian Theobalt, and Tat-Seng Chua. Learning a disentangled embedding for monocular 3d shape retrieval and pose estimation. *arXiv preprint arXiv:1812.09899*, 2018. 3
- [21] Anan Liu, Zhongyang Wang, Weizhi Nie, and Yuting Su. Graph-based characteristic view set extraction and matching for 3d model retrieval. *Information Sciences*, 320:429–442, 2015. 3
- [22] An-An Liu, Yu-Ting Su, Wei-Zhi Nie, and Mohan Kankanhalli. Hierarchical clustering multi-task learning for joint human action grouping and recognition. *IEEE transactions on pattern analysis and machine intelligence*, 39(1):102–114, 2016. 1
- [23] Zhenguang Liu, Kedi Lyu, Shuang Wu, Haipeng Chen, Yanbin Hao, and Shouling Ji. Aggregated multi-gans for controlled 3d human motion prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021. Accepted. 1
- [24] Zhenguang Liu, Shuang Wu, Shuyuan Jin, Qi Liu, Shijian Lu, Roger Zimmermann, and Li Cheng. Towards natural and accurate future motion prediction of humans and animals. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10004–10012, 2019. 1
- [25] Yue Luo, Jimmy Ren, Zhouxia Wang, Wenxiu Sun, Jinshan Pan, Jianbo Liu, Jiahao Pang, and Liang Lin. Lstm pose ma-

- chines. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5207–5215, 2018. 1, 3
- [26] Gyeongsik Moon, Ju Yong Chang, and Kyoung Mu Lee. Posefix: Model-agnostic general human pose refinement network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7773–7781, 2019. 3
- [27] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hour-glass networks for human pose estimation. In *European conference on computer vision*, pages 483–499. Springer, 2016. 1, 3
- [28] Tomas Pfister, James Charles, and Andrew Zisserman. Flowing convnets for human pose estimation in videos. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1913–1921, 2015. 2, 3
- [29] Benjamin Sapp, Alexander Toshev, and Ben Taskar. Cascaded models for articulated pose estimation. In *European conference on computer vision*, pages 406–420. Springer, 2010. 1, 2
- [30] Michael Snower, Asim Kadav, Farley Lai, and Hans Peter Graf. 15 keypoints is all you need. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6738–6748, 2020. 5
- [31] Jie Song, Limin Wang, Luc Van Gool, and Otmar Hilliges. Thin-slicing network: A deep structured model for pose estimation in videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4220–4229, 2017. 2, 3
- [32] Kai Su, Dongdong Yu, Zhenqi Xu, Xin Geng, and Changhu Wang. Multi-person pose estimation with enhanced channel-wise and spatial information. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5674–5682, 2019. 3
- [33] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5693–5703, 2019. 2, 3, 4, 5, 6, 8
- [34] Min Sun, Pushmeet Kohli, and Jamie Shotton. Conditional regression forests for human pose estimation. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3394–3401. IEEE, 2012. 2
- [35] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. Learning to compose dynamic tree structures for visual contexts. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6619–6628, 2019. 3
- [36] 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. 1
- [37] Ali Varamesh and Tinne Tuytelaars. Mixture dense regression for object detection and human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13086–13095, 2020. 3
- [38] Fang Wang and Yi Li. Beyond physical connections: Tree models in human pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 596–603, 2013. 1, 2
- [39] Manchen Wang, Joseph Tighe, and Davide Modolo. Combining detection and tracking for human pose estimation in videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11088–11096, 2020. 2, 4, 5, 6
- [40] Yang Wang and Greg Mori. Multiple tree models for occlusion and spatial constraints in human pose estimation. In *European Conference on Computer Vision*, pages 710–724. Springer, 2008. 1, 2
- [41] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 1, 3
- [42] Philippe Weinzaepfel, Jerome Revaud, Zaid Harchaoui, and Cordelia Schmid. Deepflow: Large displacement optical flow with deep matching. In *Proceedings of the IEEE international conference on computer vision*, pages 1385–1392, 2013. 3
- [43] Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking. In *Proceedings of the European conference on computer vision (ECCV)*, pages 466–481, 2018. 3, 5, 6, 8
- [44] Yuliang Xiu, Jiefeng Li, Haoyu Wang, Yinghong Fang, and Cewu Lu. Pose flow: Efficient online pose tracking. *arXiv preprint arXiv:1802.00977*, 2018. 5, 6
- [45] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015. 5
- [46] Feng Zhang, Xiatian Zhu, Hanbin Dai, Mao Ye, and Ce Zhu. Distribution-aware coordinate representation for human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7093–7102, 2020. 3
- [47] Jiabin Zhang, Zheng Zhu, Wei Zou, Peng Li, Yanwei Li, Hu Su, and Guan Huang. Fastpose: Towards real-time pose estimation and tracking via scale-normalized multi-task networks. *arXiv preprint arXiv:1908.05593*, 2019. 5, 6
- [48] Xiaoqin Zhang, Changcheng Li, Xiaofeng Tong, Weiming Hu, Steve Maybank, and Yimin Zhang. Efficient human pose estimation via parsing a tree structure based human model. In *2009 IEEE 12th International Conference on Computer Vision*, pages 1349–1356. IEEE, 2009. 1, 2
- [49] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9308–9316, 2019. 5
- [50] Yuan Zhuang, Zhenguang Liu, Peng Qian, Qi Liu, Xiang Wang, and Qinming He. Smart contract vulnerability detection using graph neural network. In *IJCAI*, pages 3283–3290, 2020. 3