Temporal Keypoint Matching and Refinement Network for Pose Estimation and Tracking

Chunluan Zhou Zhou Ren Gang Hua

Wormpex AI Research
czhou002@e.ntu.edu.sg, {renzhou200622, ganghua}@gmail.com

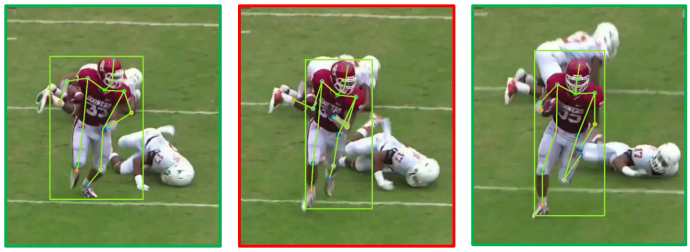
Abstract. Multi-person pose estimation and tracking in realistic videos is very challenging due to factors such as occlusions, fast motion and pose variations. Top-down approaches are commonly used for this task, which involves three stages: person detection, single-person pose estimation, and pose association across time. Recently, significant progress has been made in person detection and single-person pose estimation. In this paper, we mainly focus on improving pose association and estimation in a video to build a strong pose estimator and tracker. To this end, we propose a novel temporal keypoint matching and refinement network. Specifically, we propose two network modules, temporal keypoint matching and temporal keypoint refinement, which are incorporated into a single-person pose estimatin network. The temporal keypoint matching module learns a simialrity metric for matching keypoints across frames. Pose matching is performed by aggregating keypoint similarities between poses in adjacent frames. The temporal keypoint refinement module serves to correct individual poses by utilizing their associated poses in neighboring frames as temporal context. We validate the effectiveness of our proposed network on two benchmark datasets: PoseTrack 2017 and PoseTrack 2018. Exprimental results show that our approach achieves state-of-the-art performance on both datasets.

Keywords: Pose estimation and tracking, Temporal keypoint matching, Temporal keypoint refinement

1 Introduction

Human pose estimation and tracking aims at predicting the body parts (or keypoints) of each person in each frame of a video and associate them in the spatial-temporal space across the video. It could facilitate various applications such as augmented reality, human-machine interaction and action recognition [8, 21], and has recently gained considerable research attenion [17, 29, 11, 28, 16, 25]. Human pose estimation and tracking in videos is a very challenging task due to pose variations, scale variations, fast motion, occlusions, complex backgrounds, etc. There are mainly two categories of approaches for this task: top-down [11, 28] and bottom-up [1, 30, 29, 25]. The main difference between them is how pose estimation is performed in *single images*: bottom-up approaches detect individual part candidates in an image and group them into poses, while top-down

(a) Target drifting happens when two persons overlap



(b) Pose estimation is difficult without temporal context

Fig. 1. Issues of pose association and estimation in videos.

approaches first locate each person in the image and then perform single-person pose estimation. Considering the superior performance of top-down pose estimation approaches [7, 28, 18], in this work we explore how to build a high-quality multi-person pose estimator and tracker on top of them.

Generally, top-down pose estimation and tracking involves three stages: person detection, single-person pose estimation, and pose association across time. With the development of deep convolutional neural networks, significant progress has been made in person detection [26, 12, 31] and single-person pose estimation [7, 28, 18]. Despite the availability of advanced techniques for the first two stages, there are still two main challenges for top-down pose estimation and tracking: pose association and pose estimation in a video. For pose association across frames, target drifting often occurs due to complex intersection of multiple people in a video. For example, in Fig. 1(a), the severe occlusion and similar appearance makes it difficult to track the dancer in the purple bounding-box in the left image. For pose estimation in a video, occlusions, motion blur, distraction from other persons and complex backgrounds could greatly increase the ambiguity of keypoint localization. For example, it is difficult to predict the right elbow and wrist of the player due to occlusion as shown in Fig. 1(b). Temporal context could be helpful for resolving this problem.

To address the above challenges, we propose a novel temporal keypoint matching and refinement network for human pose estimation and tracking. Specifically, two network modules, temporal keypoint matching and temporal keypoint refinement, are designed and incorporated into a single-person pose estimation

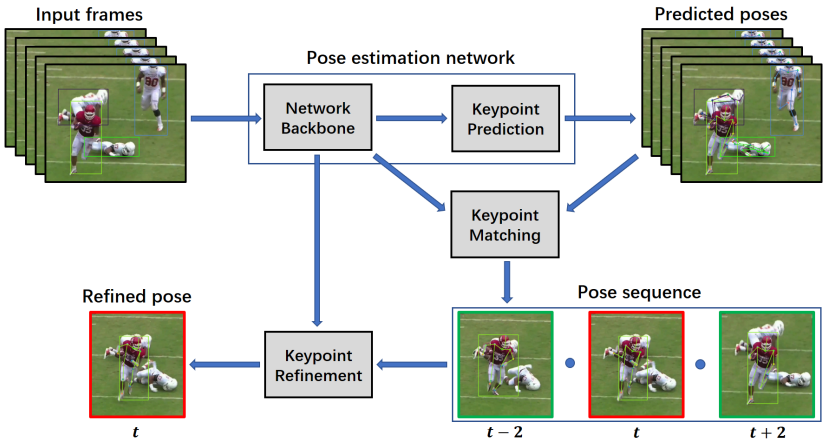


Fig. 2. Overview of our approach.

network as shown in Fig. 2. The temporal keypoint matching module learns a similarity metric for matching keypoints across frames. For pose association, two commonly used similarity metrics are intersection over union and object keypoint similarity [28]. They simply use instance-agnostic information: location or geometry. Different from them, our similarity metric is learned to distinguish keypoints from different person instances. The similarity between two poses in adjacent frames is computed by aggregating the keypoint similarities. To improve pose estimation in a video, the temporal keypoint refinement module serves to correct individual poses by utilizing its associated poses in neighboring frames as temporal context. We demonstrate the effectiveness of the proposed temporal keypoint and refinement network on two benchmark datasets: PoseTrack 2017 and PoseTrack 2018. Experimental results show that our proposed approach achieves state-of-the-art performance on both datasets.

2 Related work

2.1 Single-image pose estimation

Generally, single-image pose estimation can be classified into two categories: top-down and bottom-up. Top-down approaches first detect persons in an image and then estimate the poses for each detected person. The performance of these approaches highly rely on the quality of person detectors and single-person pose estimators. Most approaches adopt off-the-shelf detectors [26, 12, 31, 6] and focus on how to improve pose estimators [23, 7, 28, 18]. Mask R-CNN [12] integrates human detection and pose estimation in a unified network, while the majority of top-down approaches [23, 7, 28, 18] adopt separate person detector and pose estimator. The latter usually scale detected persons to a fixed large resolution, which can achieve scale invariance. As analyzed in [28, 18], large-resolution input

is beneficial for achieving better performance. Bottom-up approaches [4, 22, 16, 19] detect body parts or keypoints and group them into individual persons. Its performance relies on two components: body part detection and association. A recent trend for bottom-up approaches is to learn associative fields [4, 19] or embeddings [22, 16] for body part groupping. One major advantage of bottom-up approaches is its fast processing speed [4, 16, 19], while top-down approaches [7, 28, 18] generally have superior performance. In this work, we explore how to build a high-quality pose estimator and tracker based on top-down approaches.

2.2 Multi-person pose tracking

Multi-person pose tracking can be categorized into two classes: offline pose tracking and online pose tracking. Offline tracking approaches usually take a certain length of video frames into consieration, which allows the modelling of complex spatial-temporal relations to achieve robust tracking but usually suffers from a high computational cost. Graph partitioning based approaches [14, 17, 15] are commonly used for offline pose tracking. Online pose tracking approaches usually do not model long-range spatiotemporal relationships and are more efficient in practice. Recently, most online pose tracking approaches [16, 9, 28] adopt bipartite matching to associate poses in adjacent frames. For pose tracking with bipartite matching, the choice of similarity metric could be of great importance. The approaches in [16, 9] only utilize location or geometry information which is instance agnostic. To improve tracking robustness, both human-level and temporal instance embeddings are learned to compute the similarity between two temporal person instances [16]. Our approach also adopts bipartite matching for pose tracking. Different from [16], our approach learns keypoint-level embeddings which can be exploited for pose tracking as well as for pose refinement.

2.3 Pose estimation in videos

Several methods have been proposed for human pose estimation in videos. The flowing ConvNet [24] exploits optical flow to align features temporally across multiple frames to improve pose estimation in individual frames. In [5], a personalized video pose estimation framework is proposed to discover discriminative appearance features from adjacent frames to finetuning a single-frame person estimation network. In the work [27], a spatio-temporal CRF is incorporated into a deep convolutional neural network to utilize both spatial and temporal cues for pose prediction in a video. Recently, PoseWarper [3] is proposed to augment pose annotations for sparsely annotated videos. These approaches are mainly designed to exploit temporal context for improving single-frame pose estimation, while our approach aims to improve both pose association and estimation in a video for multi-person pose estimation and tracking.

3 Proposed approach

We propose a temporal keypoint matching and refinement network for human pose estimation and tracking. The overview of the proposed network is illustrated in Fig. 2. We design two modules, temporal keypoint matching and temporal keypoint refinement, to improve pose association and pose estimation in a video respectively. The two modules are added to a top-down pose estimation network which is comprised of a network backbone and a keypoint prediction module as shown in Fig. 2. The keypoint prediction module produces intial poses for subsequent pose association and refinement.

3.1 Single-frame pose estimation

As in [28, 18], we adopt a top-down approach for single-frame pose estimation. Each person detection is cropped from a video frame and scaled to a fixed size of $H \times W$ before being fed to the network for pose estimation. The network backbone takes the scaled person detection as input and outputs a set of feature maps. With the feature maps, the keypoint prediction module produces K heatmaps M^k for $1 \leq k \leq K$, where K is the number of pre-defined keypoints and M^k is the heatmap for the k-th keypoint. The keypoint prediction module consists of three deconvolution layers followed by a 1×1 convolution layer of K channels. Let $\bar{H} \times \bar{W}$ be the resolution of the heatmaps, where $\bar{H} = \frac{H}{s}$ and $\bar{W} = \frac{W}{s}$ with s a scaling factor. The location which has the highest response in the heatmap M^k is taken as the predicted location for the k-th keypoint:

$$l_k^* = \arg\max_l M^k(l), \tag{1}$$

where $M^k(l)$ is the response at the location l of the heatmap M_k.

To train the keypoint prediction module, person examples are cropped from training images and scaled to the size of $H \times W$. Each person example is annotated with K keypoints. Denote by $\bar{P}_i$ $(1 \leq i \leq N)$ the i-th person example and $\bar{Q}_i = \{(\bar{l}_i^k, \bar{v}_i^k) | 1 \leq k \leq K\}$ the keypoint annotations of $\bar{P}_i$, where $\bar{l}_i^k$ is the keypoint location and $\bar{v}_i^k \in \{0, 1\}$ indicates whether the keypoint is visible. The keypoint prediction module is trained by minimizing the following loss:

$$L_{\text{pose}} = \frac{1}{NK} \sum_{i=1}^{N} \sum_{k=1}^{K} \bar{v}_i^k ||M_i^k - \bar{M}_i^k||_2^2, \tag{2}$$

where M_i^k and $\bar{M}_i^k$ are the predicted and ground-truth heatmaps of the k-th keypoint on the i-th person example respectively. The ground-truth heatmap $\bar{M}_i^k$ is generated by a Gaussian distribution $\bar{M}_i^k(l) = \exp(\frac{-||l - \bar{l}_i^k|_2^2}{\sigma^2})$ with $\sigma = 3$.

3.2 Pose tracking with temporal keypoint matching

Most recent approaches [11, 28, 16] perform pose tracking by assigning IDs to person detections. For the first frame, all person detections are assigned different

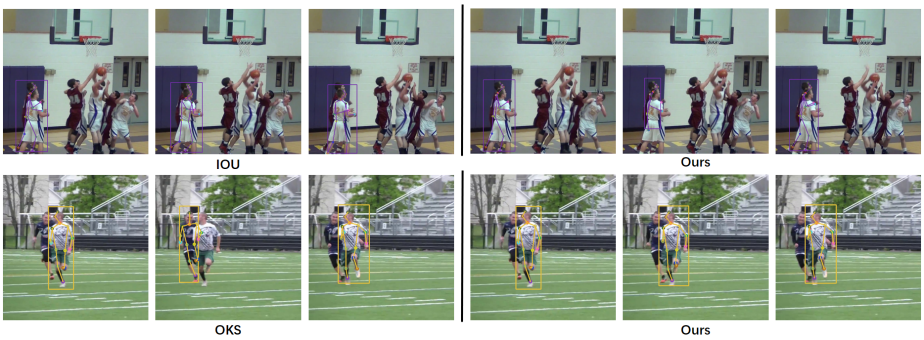


Fig. 3. Pose tracking.

IDs. Then for the following frames, person detections in frame t are matched to those in frame $t-1$. The matching is formulated by a maximum bipartite matching problem. Let $P_{t,i}$ for $1 \leq i \leq N_t$ be the i-th person detection in frame t and $w_{i,j}^{t,t-1}$ be the similarity between person detections $P_{t-1,j}$ and $P_{t,i}$. Define a binary variable $z_{i,j}^{t,t-1} \in \{0,1\}$ which indicates whether $P_{t-1,j}$ and $P_{t,i}$ are matched. The goal of maximum bipartite matching is to find the optimal solution z^*:

$$z^* = \arg\max_z \sum_{1 \leq i \leq N_t, 1 \leq j \leq N_{t-1}} z_{i,j}^{t,t-1} w_{i,j}^{t,t-1}, \tag{3}$$

$$\text{s.t. } \forall i, \sum_{1 \leq j \leq N_{t-1}} z_{i,j}^{t,t-1} \leq 1 \text{ and } \forall j, \sum_{1 \leq i \leq N_t} z_{i,j}^{t,t-1} \leq 1. \tag{4}$$

If $P_{t,i}$ is matched to $P_{t-1,j}$ ($z_{i,j}^{t,t-1} = 1$), the ID of $P_{t-1,j}$ is assigned to $P_{t,i}$. If $P_{t,i}$ is not matched to any detection in frame $t-1$, a new ID is assigned to $P_{t,i}$. Two commonly used similarity metrics for pose tracking are intersection over union (IOU) between person detections and object keypoint similarity (OKS) between poses of person detections [28]. For the IOU metric, pose tracking tends to fail when two persons are in close proximity (See Row 1 of Fig. 3). For the OKS metric, confusion could happen when the poses of two persons are similar (See Row 2 of Fig. 3). To improve the robustness of pose tracking, we propose a new similarity metric based on temporal keypoint matching.

Specifically, we abstract keypoints of person detections by feature vectors and perform keypoint matching by classification. To do this, we introduce a keypoint matching module on top of the network backbone. The module extracts features for keypoints and determines if two keypoints of the same type in spatiotemporal space belong to the same person. For a pair of temporal keypoints of the same type, the module outputs a similarity score. We define the similarity between two person detections $P_{t,i}$ and $P_{t-1,j}$ by aggregating keypoint similarities

$$w_{i,j}^{t,t-1} = I(IOU(P_{t,i}, P_{t-1,j}) \geq 0.1) \sum_{k=1} G_k(f_{t,i}^k, f_{t-1,j}^k), \tag{5}$$

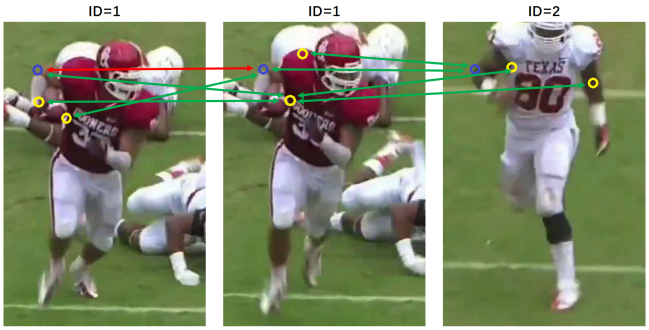


Fig. 4. Keypoint pair sampling. Blue circles are ground-truth locations and yellow circles are local maxima candidates from heatmaps. The Red line indicates a positive keypoint pair and green lines represent negative keypoint pairs.

where I is an indicator function, IOU computes the IOU between $P_{t,i}$ and $P_{t-1,j}$, $f^k_{t,i}$ is the feature vector of the k-th keypoint of $P_{t,i}$ and G_k is the similarity function for the k-th keypoint. The overlap constraint is used to avoid matching two person detections which are far away in adjacent frames. As shown in Figure 3, the key-point matching method can improve the robustness of pose tracking, especially in the situation of heavy occlusions. When some keypoints of a person are occluded, pose association can rely on the remaining visible keypoints.

The temporal keypoint matching module is implemented by a sequence of four basic blocks and K classifiers. Each basic block consists of three 3×3 convolution layers of 256 channels and a deconvolution layer which upsamples the output by a factor of 2. The four basic blocks output a set of feature maps which have the same resolution as the heatmaps (i.e. $\bar{H} \times \bar{W}$). Denote by $F_{t,i}$ the feature maps for the person detection $P_{t,i}$. The k-th keypoint $p^k_{t,i}$ of $P_{t,i}$ is represented by $F_{t,i}(p^k_{t,i})$, where $F_{t,i}(p^k_{t,i})$ is the feature vector at $p^k_{t,i}$ of $F_{t,i}$. The classifier G_k takes the concatenation of $f^k_{t,i}$ and $f^k_{t-1,j}$, and outputs a probability of the two keypoints $p^k_{t,i}$ and $p^k_{t-1,j}$ belonging to the same person. Each classifer G_k is implemented by three fully connected layers followed by a softmax layer. The first two layers have 256 output units and the third one has 2 output units.

To train the keypoint matching module, we sample a set of person examples among which some have identical IDs and the others have different IDs. Specifially, for a person example in frame t, we collect some person examples from the temporal window $[t-\tau, t+\tau]$. For each pair of person examples, we sample a set of keypoint pairs for each type of keypoint. Figure 4 illustrates keypoint pair sampling for the right elbow. For each person example, we sample some local maxima candidates in the heatmap of the right elbow. Non-maximum suppression is applied to sample sparse locations. The ground-truth location (Blue circle) is always sampled as the first location. Only the grund-truth locations of a person example pair which have the same ID is labeled as 1. The other location

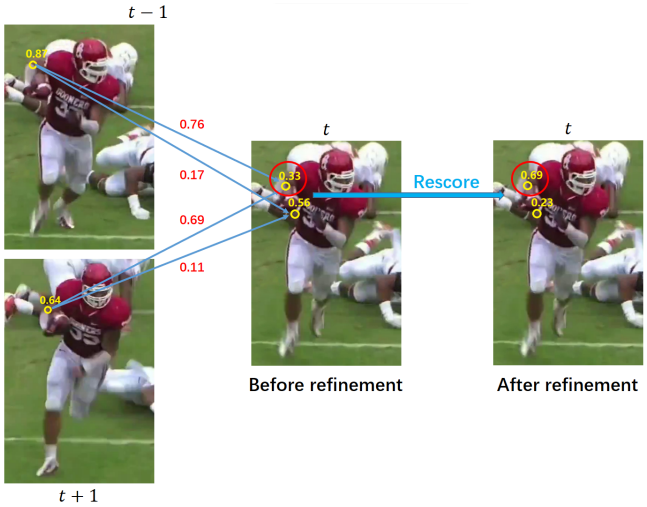


Fig. 5. Keypoint refinement. Circles are local maxima candidates sampled from heatmaps for the right elbow. Yellow numbers are responses and Red numbers are similarities. The correct right elbow locations in frames $t-1$ and $t+1$ give more support to their correct match inside the red circle in frame t. After refinement, the correct right elbow location in frame t gets a larger response than the wrong one.

pairs are labeled as 0. We keep the ratio of positive location pairs to negative location pairs to $1:4$. The cross-entropy loss is used to train the K classifiers.

3.3 Pose refinement with temporal context

Pose estimation based on a single frame in a video could be very challenging, due to factors like occlusions, distraction of keypoints from other persons, motion blur, clutters, etc. As shown in Figure 5, the correct location of the right elbow of the person detection in frame t has a lower response than the other local maxima candidate since the right elbow is partially occluded. When looking at the neighboring frames $t-1$ and $t+1$, we can find that the right elbow is still visible and correctly predicted. These correctly predicted counterparts in neighboring frames could provide useful cues to correct the prediction in frame t. Motivated by this observation, we propose a method to refine the predicted pose of a person detection in a frame by exploiting its counterparts in neighboring frames as temporal context.

For each person detection $P_{t,i}$ in frame t, we search for its counterparts in a temporal window $[t-\tau, t+\tau]$. Specifically, we search for two paths in backward and forward directions respectively. For the backward path search, we start the path at the person detection $P_{t,i}$ in frame t. Then, the person detection in frame $t-1$ that has the highest similarity to $P_{t,i}$ according to Eq. (5) is selected. Then, the selected person detection in frame $t-1$ is taken as the reference and the best matching person detection in $t-2$ is obtained. This process continues until

frame $t-\tau$ is reached. Similarly, the forward path search is performed in the opposite direction. Next, we merge the two paths into a single path. The person detections on this path are used to refine the predicted pose of $P_{t,i}$.

Let $\mathcal{Q} = \{P_{t',i} | t-\tau \leq t' \leq t+\tau\}$ be the set of person detections on the selcted path of a certain $P_{t,i}$. Denote by $M^k_{t,i}$ the heatmap of the k-th keypoint of the detection $P_{t,i}$. For the k-th keypoint, we refine the heatmap $M^k_{t,i}$. For this purpose, we propose a keypoint refinement module which has the same structure as the keypoint prediction module but is applied in a different way. Specifically, we sparsely sample a set of n local maxima candidates in $M^k_{t,i}$ and refine their responses. We set $n = 16$ in our experiments and find it sufficient to cover most correct locations of all types of keypoints.

Let $\mathcal{L}^k_{t,i}$ be the set of n local maxima candidates of the k-th keypoint on person detection $P_{t,i}$. We take the predicted locations of the k-th keypoint on other counterparts in $\mathcal{Q}$ as the context for the local maxima candidates in $\mathcal{L}^k_{t,i}$. Denote by $\hat{l}^k_{t',i}$ the location with the highest response in $M^k_{t',i}$. We use the output of the last deconvolution layer in the keypoint refinement module as features to represent person detections for pose refinement. Let $H_{t,i}$ be the feature maps of the person detection $P_{t,i}$. To refine the response at l for the k-th keypoint, we aggregate the feature vector $H_{t,i}(l)$ and feature vectors $H_{t',i}(\hat{l}^k_{t',i})$ for $t' \in [t-\tau, t+\tau] \setminus t$ by

$$\bar{H}_{t,i}(l) = \frac{H_{t,i}(l) + \sum_{t' \in [t-\tau,t+\tau]\setminus t} H_{t',i}(\hat{l}^k_{t',i}) W(l, \hat{l}^k_{t',i})}{2\tau + 1}, \tag{6}$$

where $W(l, \hat{l}^k_{t',i})$ is the similarity between l and $\hat{l}^k_{t',i}$ output by the keypoint matching module. For keypoint refinement, the aggregated feature vector $\bar{H}_{t,i}(l)$ instead of the original one $H_{t,i}(l)$ is taken as input to produce a new response. With $W(l, \hat{l}^k_{t',i})$ as a weight, $\hat{l}^k_{t',i}$ gives more support to its correct matching location. Figure 5 illustrates the proposed keypoint refinement method for the right elbow. The keypoint refinement module is trained using the same loss as in Eq. (2) except that only a sparse set of candidate locations are used to update the loss during back-propagation.

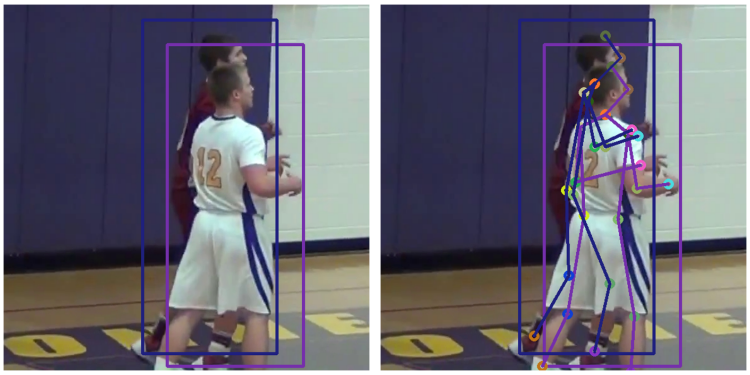
3.4 Training

We adopt a two-stage training procedure. In the first stage, we train a single-frame pose estimation model as described in Section 3.1. In the second stage, we use the model trained in the first stage to initialize our network and fix the weights of the backbone and keypoint prediction module during model optimization. Stochastic gradient descent is adopted for updating model weights.

4 Experiments

4.1 Datasets and evaluation

We evaluate our approach on two recently published large-scale benchmark datasets: PoseTrack 2017 and PoseTrack 2018 [1], for multi-person pose esti-



(a) Person detections (b) Predicted poses

Fig. 6. Pose-based NMS.

mation and tracking. The PoseTrack 2017 dataset contains 250 video clips for training and 50 video clips for validation. The size of the PoseTrack 2018 dataset is doubled. For PoseTrack 2018, we also use the train split for training and the validation split for testing. It is a common practice to use either COCO [20] or MPII [2] for model pre-training [1]. In our experiments, we use the COCO dataset to pre-train single-frame pose estimation models used in our experiments. Following [1], we use average precision (AP) to measure the multi-person pose estimation performance and multi-object tracking accuracy (MOTA) to measure the tracking performance.

4.2 Implementation details

We follow [28] to train single-frame pose estimation models. Two network backbones, Resnet-152 [13] and HRNet [18], are used in our experiments. For single-frame model training, we iterate 20 epochs. The inital learning rate is set to 0.001 at the beginning and is reduced two times by a factor of 10 at 10 and 15 epochs, respectively. For training the keypoint matching module and keypoint refinement module, we set the length of temporal window to 11 (i.e. $\tau = 5$). The model is trained for 9 epochs. The intial learning rate is set to 0.0001 and is reduced by a factor of 10 at epoch 7. We use Faster R-CNN [26] with feature pyramid network (FPN) and deformable convolutional network (DCN) to train our detectors. The detectors are also pre-trained on COCO and fine-tuned on PoseTrack 2017 and PoseTrack 2018, respectively.

For the first stage of multi-person pose estimation and tracking, non-maximum suppresion (NMS) is commonly applied to remove duplicate detections. As multile people in a video often involve complex interaction, person-to-person occlusions occur frequently. The conventional NMS based on bounding-box intersection over union (IOU) is prone to fail when two people are in close proximity as shown in Fig. 6(a). As the detection results could affect the subsequent pose

Method	Head	Shou	Elb	Wri	Hip	Knee	Ankl	Total
BUTD [17]	79.1	77.3	69.9	58.3	66.2	63.5	54.9	67.8
RPAF [30]	83.8	84.9	76.2	64.0	72.2	64.5	56.6	72.6
ArtTrack [1]	78.7	76.2	70.4	62.3	68.1	66.7	58.4	68.7
PoseFlow [29]	66.7	73.3	68.3	61.1	67.5	67.0	61.3	66.5
STAF [25]	-	-	-	65.0	-	-	62.7	72.6
ST-Embed [16]	83.8	81.6	77.1	70.0	**77.4**	74.5	70.8	77.0
DAT [11]	67.5	70.2	62.0	51.7	60.7	58.7	49.8	60.6
FlowTrack [28]	81.7	83.4	**80.0**	**72.4**	75.3	74.8	67.1	76.9
Ours	**85.3**	**88.2**	79.5	71.6	76.9	**76.9**	**73.1**	**79.5**
PoseWarper* [3]	81.4	88.3	83.9	78.0	82.4	80.5	73.6	81.2

Table 1. Comparison with state-of-the-art on single-frame pose estimation on PoseTrack 2017 Validation. Numbers in the table refer to mAP. "*" means that unlabelled frames are exploited for training and no threshold is used to filter keypoints for evaluation.

Method	Head	Shou	Elb	Wri	Hip	Knee	Ankl	Total
BUTD [17]	71.5	70.3	56.3	45.1	55.5	50.8	37.5	56.4
ArtTrack [1]	66.2	64.2	53.2	43.7	53.0	51.6	41.7	53.4
PoseFlow [29]	59.8	67.0	59.8	51.6	60.0	58.4	50.5	58.3
STAF [25]	-	-	-	-	-	-	-	62.7
ST-Embed [16]	78.7	79.2	**71.2**	61.1	**74.5**	69.7	**64.5**	71.8
DAT [11]	61.7	65.5	57.3	45.7	54.3	53.1	45.7	55.2
FlowTrack [28]	73.9	75.9	63.7	56.1	65.5	65.1	53.5	65.4
Ours	**81.0**	**82.9**	69.8	**63.6**	72.0	**71.1**	60.8	**72.2**

Table 2. Comparison with state-of-the-art methods on multi-person pose tracking on PoseTrack 2017 Validation. Numbers in the table refer to MOTA.

estimation, tracking and refinement, we implement a simple variant of the pose based NMS (pNMS) [10] to better handle occlusions for person detection, as illustrated in 6(b). For two person detections, we compare their poses by computing the distance of each keypoint pair. If the distance within a threshold, the two keypoints are considered to be identical. Then, we count how many keypoint pairs coincide in the two poses. If the percentage is larger than 0.5, we determine that the two person detections correspond to the same person.

4.3 Results on PoseTrack 2017

Comparison with state-of-the-art We compare our approach with state-of-the-art multi-person pose estimation and tracking approaches in Tables 1 and 2. The first six approaches are bottom-up approaches while the remaining three are top-down approaches.

Table 1 shows the results of single-frame pose estimation on the PoseTrack 2017 validation subset. Our approach outperforms the most competitive top-down approach, FlowTrack [28], by 2.6%, and outperforms the best bottom-up

Method	Backbone	Detector	NMS	Similarity	Refinement	Context	mAP	MOTA
M1	ResNet-152	ResNet-101	cNMS	IOU			75.2	63.5
M2	ResNet-152	ResNet-101	pNMS	IOU			76.1	64.8
M3	ResNet-152	ResNet-101	pNMS	OKS			76.1	65.0
M4	ResNet-152	ResNet-101	pNMS	TBM			76.1	66.0
M5	ResNet-152	ResNet-101	pNMS	TKM			76.1	67.3
M6	ResNet-152	ResNet-101	pNMS	TKM	✓		76.8	68.1
M7	ResNet-152	ResNet-101	pNMS	TKM	✓	✓	78.2	69.6
M8	ResNet-152	ResNeXt-101	pNMS	TKM	✓	✓	79.0	71.4
M9	HRNet	ResNeXt-101	pNMS	TKM	✓	✓	**79.5**	**72.2**

Table 3. Ablation study on the PoseTrack 2017 validation dataset. cNMS represents the conventional IOU-based NMS. TBM uses the similarity between feature vectors of the whole bodies. Context indicates whether temporal frames are used for pose refinement.

approach, ST-Embed [16], by 2.5%. We also include the result of PoseWarper [3] in the table, as PoseWarper achieves the best performane for pose estimation on the validation subset. PoseWarper can exploit unlabeled frames for training and does not use a threshold to filter keypoints for evaluation. These are different from the common training and evaluation practice in the PoseTrack benchmark and can bring some performance improvement.

Table 2 shows the results of multi-person pose tracking. Our approach also achieves the state-of-the-art performance. Our approach improves the performance over FlowTrack significantly by 6.8%, showing that the proposed keypoint matching and refinement modules are effective for improving top-down human pose estimation and tracking. Compared to the best bottom-up approach ST-Embed, our approach achieves an improvement of 0.4% in MOTA. The improvement is not large, because ST-Embed also adopts an instance-aware similarity metric for pose tracking. The difference is that ST-Embed uses both human-level and temporal instance embeddings, while our similarity metric only uses keypoint-level embeddings.

Ablation study Table 3 shows an ablation study of our proposed approach. We compare our full model with several variants of our method, explained as follows.

The first method M1 is a re-implementation of FlowTrack [28] with two differences: (1) we do not use flow propagation to augment detections; (2) we use ResNet-101 instead of ResNet-152 to train a detector. M1 uses the conventional NMS. With the simple pNMS, the method M2 improves the performance over M1 by 0.9% in mAP and 1.3% in MOTA respetively. The pNMS can reduce the risk of suppressing ture person detections when person-to-person occlusions happen frequently. It is often the case in the PoseTrack 2017 dataset, as there are many persons appearing in a large portion of video clips. Better detection results can benefit single-frame pose estimation as well as pose tracking.

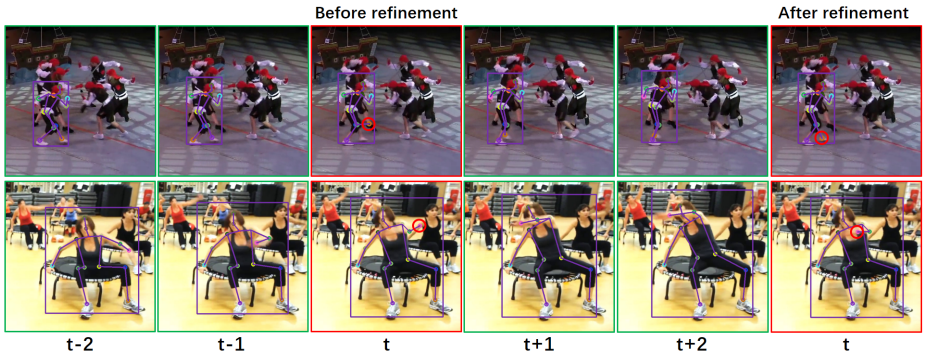


Fig. 7. Qualitative examples of keypoint refinement. Red circles indicate the keypoints which are corrected after keypoint refinement.

To demonstrate the effectiveness of our similarity metric based on temporal keypoint matching (TKM), we compare it with two commonly used similarity metrics, IOU and OKR, for pose tracking. The results of M2, M3, and M5 show that IOU and OKR achieve similar perfomance, while our propoesd TKM improves the tracking performance over IOU and OKR by over 2%. We further compare our proposed TKM with a variant (M4) in which feature vectors are learned to represent human bodies instead of keypoints. M4 improves the tracking performance over M2 and M3 by about 1%, but its performance decreases by 1.3% compared with M5. Matching persons by keypoint similarity instead of body similarity can improve the robustness of tracking especially when occlusions happen.

Next, we experiment with two pose refinement approaches. Both M6 and M7 use our propoesd keypoint refinement module for pose correction. The difference is that M6 does not use temporal frames as context but M7 does. M6 can be considered as self-refinement. Recall that we sample local maxima candidates which are then rescored by the keypont refinement module. These local maxima candidates except for true keypoint locations can be considered as hard negatives. We can see that compared to M5, both M6 and M7 improve the performce for single-image pose estimation. As a result, the performance of pose tracking is also improved. M7 further improves the performance over M6 by 1.4% in mAP and 1.5% in MOTA, respectively, showing that temporal context is helpful for correcting wrong keypoint predictions. Figure 7 shows two qualitative examples of our keypoint refinement module.

We also experiment with a stronger detector backbone, ResNeXt-101. Compared to M7, the performane is further improved by 1.2% in mAP and 1.8% in MOTA (See M8). Finally, we replace ResNet-152 with a stronger pose network backbone, HRNet. The results are pushed to 79.5% in mAP and 72.2 in MOTA.

Method	Backbone	Detector	NMS	Similarity	Refine	Context	mAP	MOTA
STAF [25]	VGG	-	-	-			70.4	60.9
N1	HRNet	ResNeXt-101	cNMS	IOU			74.1	63.7
N2	HRNet	ResNeXt-101	pNMS	IOU			74.8	65.3
N3	HRNet	ResNeXt-101	pNMS	TBM			74.8	65.9
N4	HRNet	ResNeXt-101	pNMS	TKM			74.8	67.0
N5	HRNet	ResNeXt-101	pNMS	TKM	✓		75.7	67.8
N6	HRNet	ResNeXt-101	pNMS	TKM	✓	✓	**76.7**	**68.9**

Table 4. Results on the PoseTrack 2018 validation dataset. cNMS represents the conventional IOU-based NMS. cNMS represents the conventional IOU-based NMS. TBM uses the similarity between feature vectors of the whole bodies. Context indicates whether temporal frames are used for pose refinement.

4.4 Results on PoseTrack 2018

Only one existing method STAF [14] has reported results on PoseTrack 2018 dataset. Table 4 shows the results of our approach and STAF. STAF is a bottom-up appraoch which uses a weaker network backbone VGG. Its strength lines in its real-time processing speed. We report its results in the table for reference. For the experiments on PoseTrack 2018, we only use HRNet and ResNeXt-101 as the detector and pose estimation backbones respectively. For Table 4, we can see that the simple pNMS improves the performane over conventional NMS by 0.7% in mAP and 1.6% in MOTA. With temporal keypoint matching for pose tracking, further improvment of 1.7% in MOTA is achieved (N4 vs N2). Better performance is achieved with keypoint similarity than body similarity (N4 vs N3), which demonstrates that the keypoint matching method is more robust. Equipped with the proposed keypoint matching module, the performce of our approach is pushed to 76.7% in mAP and 68.9% in MOTA. These results validate the effectivness of the designs in our approach for top-down human pose estimation and tracking.

5 Conclusion

In this paper, we propose a temporal keypoint matching and refinement network for multi-person pose estimation and tracking. We design two network modules for improving pose association and estimation in videos respectively. The two network models are incorporated into a single-person pose estimation network. The temporal keypoint matching module learns similarity metrics which are aggregated for person tracking across frames. The temporal keypoint module exploits temporal context to correct intial poses predicted by the pose estimation network. The experiments on PoseTrack 2017 and 2018 validate the superiority of our approach.

Acknowledgement. Gang Hua was supported partly by National Key R&D Program of China Grant 2018AAA0101400 and NSFC Grant 61629301.

References

1. Andriluka, M., Iqbal, U., Insafutdinov, E., Pishchulin, L.: Posetrack: A benchmark for human pose estimation and tracking. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
2. Andriluka, M., Pishchulin, L., Gehler, P., Schiele, B.: 2d human pose estimation: New benchmark and state of the art analysis. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2014)
3. Bertasius, G., Feichtenhofer, C., Tran, D., Shi, J., Torresani, L.: Learning temporal pose estimation from sparsely-labeled videos. In: Advances in Neural Information Processing Systems (NIPS) (2020)
4. Cao, Z., Simon, T., Wei, S.E., Sheikh, Y.: Realtime multi-person 2d pose estimation using part affinity fields. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
5. Charles, J., Pfister, T., Magee, D., Hogg, D., Zisserman, A.: Personalizing human video pose estimation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
6. Chen, K., Pang, J., Wang, J., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Shi, J., Ouyang, W., Loy, C.C., Lin, D.: Hybrid task cascade for instance segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
7. Chen, Y., Wang, Z., Peng, Y., Zhang, Z., Yu, G., Sun, J.: Cascaded pyramid network for multi-person pose estimation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
8. Cheron, G., Laptev, I., Schmid, C.: P-cnn: Pose-based cnn features for action recognition. In: International Conference on Computer Vision (ICCV) (2015)
9. Doering, A., Iqbal, U., Gall, J.: Jointflow: Temporal flow fields for multi person pose tracking. In: British Machine Vision Conference (BMVC) (2018)
10. Fang, H., Xie, S., Tai, Y., Lu, C.: Rmpe: Regional multi-person pose estimation. In: International Conference on Computer Vision (ICCV) (2017)
11. Girdhar, R., Gkioxari, G., Torresani, L., Paluri, M., Tran, D.: Detect-and-track: Efficient pose estimation in videos. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
12. He, K., Gkioxari, G., Dollar, P., Girshick, R.: Mask r-cnn. In: International Conference on Computer Vision (ICCV) (2017)
13. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
14. Insafutdinov, E., Andriluka, M., Pishchulin, L., Tang, S.: Arttrack: Articulated multi-person tracking in the wild. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
15. Iqbal, U., Milan, A., Gall, J.: Posetrack: Joint multi-person pose estimation and tracking. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
16. Jin, S., Liu, W., Ouyang, W., Qian, C.: Multi-person articulated tracking with spatial and temporal embeddings. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
17. 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)
18. Ke, S., Xiao, B., Liu, D., Wang, J.: Deep high-resolution representation learning for human pose estimation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)

19. Kreiss, S., Bertoni, L., Alahi, A.: Pifpaf: Composite fields for human pose estimation. In: International Conference on Computer Vision (ICCV) (2019)
20. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollar, P., Zitnick, L.: Microsoft coco: Common objects in context. In: European Conference on Computer Vision (ECCV) (2014)
21. Liu, M., Yuan, J.: Recognizing human actions as the evolution of pose estimation maps. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
22. Newell, A., Huang, Z., Deng, J.: Associative embedding: End-to-end learning for joint detection and grouping. In: Advances in Neural Information Processing Systems (NIPS) (2017)
23. Newell, A., Yang, K., Deng, J.: Stacked hourglass networks for human pose estimation. In: European Conference on Computer Vision (ECCV) (2016)
24. Pfister, T., Charles, J., Zisserman, A.: Flowing convnets for human pose estimation in videos. In: International Conference on Computer Vision (ICCV) (2015)
25. Raaj, Y., Idrees, H., Hidalgo, G., Sheikh, Y.: Efficient online multi-person 2d pose tracking with recurrent spatio-temporal affinity fields. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
26. 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 (NIPS) (2015)
27. Song, J., Wang, L., Van Gool, L., Hilliges, O.: Thin-slicing network: A deep structural model for pose estimation in videos. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
28. Xiao, B., Wu, H., Wei, Y.: Simple baselines for human pose estimation. In: European Conference on Computer Vision (ECCV) (2018)
29. Xiu, Y., Li, J., Wang, H., Fang, Y., Lu, C.: Pose flow: Efficient online pose tracking. In: British Machine Vision Conference (BMVC) (2018)
30. Zhu, X., Jiang, Y., Luo, Z.: Multi-person pose estimation for posetrack with enhanced part affinity fields. In: ICCV PoseTrack Workshop (2017)
31. Zhu, X., Hu, H., Lin, S., Dai, J.: Deformable convnets v2: More deformable, better results. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)