Self-supervised Keypoint Correspondences for Multi-Person Pose Estimation and Tracking in Videos

Umer Rafi[1]⋆, Andreas Doering[1]⋆, Bastian Leibe[2], and Juergen Gall[1]

[1] University of Bonn, Germany
[2] RWTH Aachen, Germany

Abstract. Video annotation is expensive and time consuming. Consequently, datasets for multi-person pose estimation and tracking are less diverse and have more sparse annotations compared to large scale image datasets for human pose estimation. This makes it challenging to learn deep learning based models for associating keypoints across frames that are robust to nuisance factors such as motion blur and occlusions for the task of multi-person pose tracking. To address this issue, we propose an approach that relies on keypoint correspondences for associating persons in videos. Instead of training the network for estimating keypoint correspondences on video data, it is trained on a large scale image dataset for human pose estimation using self-supervision. Combined with a top-down framework for human pose estimation, we use keypoint correspondences to (i) recover missed pose detections and to (ii) associate pose detections across video frames. Our approach achieves state-of-the-art results for multi-frame pose estimation and multi-person pose tracking on the PoseTrack 2017 and 2018 datasets.

1 Introduction

Human pose estimation is a very active research field in computer vision that is relevant for many applications like computer games, security, sports, and autonomous driving. Over the years, the human pose estimation models have been greatly improved [12, 32, 7, 23, 42, 4, 25] due to the availability of large scale image datasets for human pose estimation [27, 2, 41]. More recently, researchers started to tackle the more challenging problem of multi-person pose tracking [20, 19, 42, 38, 46].

In multi-person pose tracking, the goal is to estimate human poses in all frames of a video and associate them over time. However, video annotations are costly and time consuming. Consequently, recently proposed video datasets [1] are less diverse and are sparsely annotated as compared to large scale image datasets for human pose estimation [27, 41]. This makes it challenging to learn deep networks for associating human keypoints across frames that are robust to
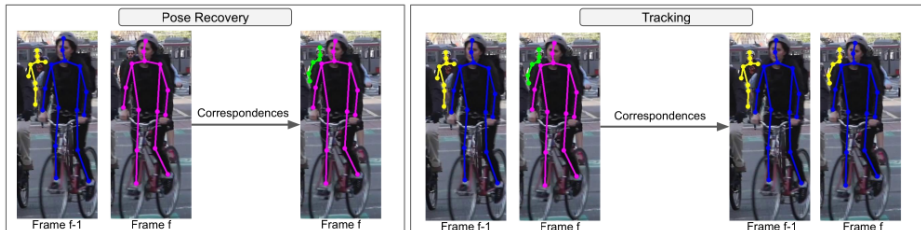

⋆ equal contribution

Fig. 1. Our contributions: (Left) We use keypoint correspondences to recover missed pose detections by using the temporal context of the previous frame. (Right) We use keypoint correspondences to associate detected and recovered pose detections for the task of multi-person pose tracking.

nuisance factors such as motion blur, fast motions, and occlusions as they occur in videos.

State-of-the-art approaches [42, 38, 14] rely on optical flow or person re-identification [46] in order to track the persons. Both approaches, however, have disadvantages. Optical flow fails if a person becomes occluded which results in a lost track. While person re-identification allows to associate persons even if they disappeared for a long time, it remains difficult to associate partially occluded persons with person re-identification models that operate on bounding boxes of the full person. Moreover, the limited annotations in pose tracking datasets require to train the models on additional datasets for person re-identification.

We therefore propose to learn a network that infers keypoint correspondences for multiple persons. The correspondence network comprises a Siamese matching module that takes a frame with estimated human poses as input and estimates the corresponding poses for a second frame. Such an approach has the advantage that it is not limited to a fixed temporal frame distance, and it allows to track persons when they are partially occluded. Our goal is to utilize keypoint correspondences to recover missed poses of a top-down human pose estimator, *e.g.*, due to partial occlusion, and to utilize keypoint correspondences for multi-person tracking. The challenge, however, is to train such a network due to the sparsely annotated video datasets. In fact, in this work we consider the extreme case where the network is not trained on any video data or a dataset where identities of persons are annotated. Instead we show that such a network can be trained on an image dataset for multi-person pose estimation [27]. Besides of the human pose annotations, which are anyway needed to train the human pose estimator, the approach does not require any additional supervision. In order to improve the keypoint associations, we propose an additional refinement module that refines the affinity maps of the Siamese matching module.

Table 1. Overview of related works on multi-person pose tracking.

Method	Detection Improvement	Tracking
Ours	Correspondences	Keypoint Correspondences
HRNet [38]	Temporal OKS	Optical Flow
POINet [36]	-	Ovonic Insight Net
MDPN [14]	Ensemble	Optical FLow
LightTrack [33]	Ensemble / BBox Prop.	GCN
ProTracker [13]	-	IoU
STAF [35]	-	ST Fields
ST Embeddings [21]	-	ST Embeddings
JointFlow [11]	-	Flow Fields

To summarize, the contributions of the paper are:

– We propose an approach for multi-frame pose estimation and multi-person pose tracking that relies on self-supervised keypoint correspondences which are learned from a large scale image dataset with human pose annotations.
– Combined with a top-down pose estimation framework, we use keypoint correspondences in two ways as illustrated in Figure 1: We use keypoint correspondences to (i) recover pose detections that have been missed by the top-down pose estimation framework and to (ii) associate detected and recovered poses in different frames of a video.
– We evaluate the approach on the PoseTrack 2017 and 2018 datasets for the tasks of multi-frame pose estimation and multi-person pose tracking. Our approach achieves state-of-the-art results without using any additional training data except of [27] for the proposed correspondence network.

2   Related Work

Multi-Person Pose Estimation. Multi-person pose estimation can be categorized into top-down and bottom-up approaches. Bottom-up based methods [23, 7, 30, 16, 32] first detect all person keypoints simultaneously and then associate them to their corresponding person instances. For example, Chao *et al.* [7] predict part affinity fields which provide information about the location and orientation of the limbs. For the association, a greedy approach is used. More recently, Kocabas *et al.* [23] propose to detect bounding boxes and pose keypoints within the same neural network. In the first stage, bounding box predictions are used to crop from predicted keypoint heatmaps. As a second stage, a pose residual module is proposed, which regresses the respective keypoint locations of each person instance.

Top-down methods [44, 25, 8, 29, 42, 44, 31] utilize person detectors and estimate the pose on each image crop individually. In contrast to bottom-up methods, top-down approaches do not suffer from scale variations. For example, Xiao *et al.* [42] propose a simple yet strong model based on a ResNet152 [17] and achieve state-of-the-art performance by replacing the last fully connected layer

by three transposed convolutions. Li *et al.* [25] propose an information prop-
agation procedure within a multi-stage architecture based on four ResNet50
networks [17] with coarse-to-fine supervision.

Multi-Frame Pose Estimation. In video data, such as PoseTrack [1], related
works [43, 14, 4] leverage temporal information of neighboring frames to increase
robustness against fast motions, occlusion, and motion blur. Xiu *et al.* [43] and
Guo *et al.* [14] utilize optical flow to warp preceding frames into the current
frame. Recently, Bertasius *et al.* [4] propose a feature warping method based on
deformable convolutions to warp pose heatmaps from preceding and subsequent
frames into the current frame. While they show that they are able to learn from
sparse video annotations, they do not address multi-person pose tracking.

Multi-Person Pose Tracking. Early works for multi-person pose tracking [20, 19]
build spatio-temporal graphs which are solved by integer linear programming.
Since such approaches are computationally expensive, researchers reduced the
task to bipartite graphs which are solved in a greedy fashion [38, 36, 14, 33, 13, 11,
35, 43, 21]. Girdhar *et al.* [13] propose a 3D Mask R-CNN [16] to generate person
tubelets for tracking which are associated greedily. More recent works [42, 38, 14,
46] incorporate temporal information by using optical flow. Xiao *et al.* [42] rely
on optical flow to recover missed person detections and propose an optical-flow
based similarity metric for tracking. In contrast, Zhang *et al.* [46] builds on [13]
and propose an adapted Mask R-CNN [16] with a greedy bounding box genera-
tion strategy. Furthermore, optical flow and a person re-identification module are
combined for tracking. Jin *et al.* [21] perform multi-person pose estimation and
tracking within a unified framework based on human pose embeddings. Table 1
provides a summary of the contributions of the recent related works.

Correspondences. In recent years, deep learning has been successfully applied
to the task of correspondence matching [9, 22, 15], including the task of visual
object tracking (VOT) [5, 24, 47, 48]. All of the above approaches establish corre-
spondences at object level. In contrast, our approach establishes correspondences
at instance level. Moreover, VOT tasks assume a ground truth object location
for the first frame, which is in contrast to the task of pose tracking.

Self-Supervised Learning. Self-supervised learning approaches [40, 26] have been
proposed for establishing correspondences at patch and keypoints level from
videos. However, these approaches use videos for learning and process a sin-
gle set of keypoints or patch at a time. In contrast, our approach establishes
correspondences for multiple instances and is trained on single images.

3   Method Overview

In this work, we propose a multi-person pose tracking framework that is robust
to motion blur and severe occlusions, even though it does not need any video

Fig. 2. Given a sequence of frames, we detect a set of person bounding boxes and perform top-down pose estimation. Our proposed method uses keypoint correspondences to (i) recover missed detections and (ii) to associate detected and recovered poses to perform tracking. The entire framework does not require any video data for training since the network for estimating keypoint correspondences is trained on single images using self-supervision.

data for training. As it is illustrated in Figure 2, we first estimate for each frame the human poses and then track them. For multi-person human pose estimation, we utilize an off-the-shelf object detector [6] to obtain a set of bounding boxes for the persons in each frame. For each bounding box, we then perform multi-person pose estimation in a top-down fashion by training an adapted GoogleNet [39], which we will discuss in Section 6.1.

In order to be robust to motion blur and severe occlusions, we do not use optical flow in contrast to previous works like [42]. Instead we propose a network that estimates for a given frame with estimated keypoints the locations of the keypoints in another frame. We use this network for recovering human poses that have been missed by the top-down pose estimation framework as described in Section 5.1 and for associating detected and recovered poses across the video as described in Section 5.2.

The main challenge for the keypoint correspondence network is the handling of occluded keypoints and the limited amount of densely annotated video data. In order to address these issues, we do not train the network on video data, but on single images using self-supervision. In this way, we can simulate disappearing keypoints by truncation and leverage large scale image dataset like MS-COCO [27] for tracking. We will first describe the keypoint correspondence network in Section 4 and then discuss the tracking framework in Section 5.

4   Keypoint Correspondence Network

Given two images I_1 and I_2 with keypoints $\{j^p\}_{1:N_p}$ for all persons p in image I_1, our goal is to find the corresponding keypoints in I_2. Towards this end, we use a Siamese network as shown in Figure 3 which estimates for each keypoint an affinity map. The affinity maps are further improved by the refinement module, which is described in Section 4.2.
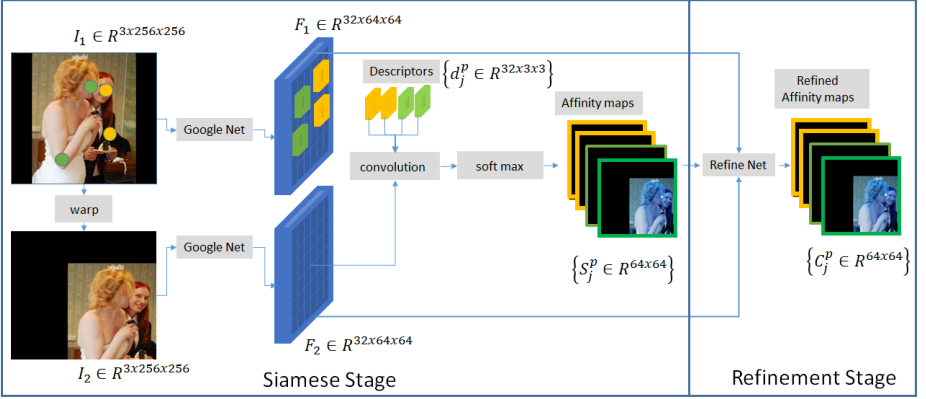
Fig. 3. Keypoint correspondence network. The Siamese network takes images I_1 and I_2 and keypoints $\{j^p\}_{1:N_p}$ for all persons p in image I_1 as input and generates the feature maps F_1 and F_2, respectively. The keypoints of the different persons are shown in green and yellow, respectively. For each keypoint, a descriptor d_j^p is extracted from F_1 and convolved with the feature map F_2 to generate an affinity map S_j^p. In order to improve the affinity maps for each person, the refinement network takes F_1, F_2 and the affinity maps S_j^p for person p as input and generates refined affinity maps C_j^p.

4.1   Siamese Matching Module

The keypoint correspondence network consists of a Siamese network. Each branch in the Siamese network is a batch normalized GoogleNet up to layer 17 with shared parameters [39]. The Siamese network takes an image pair (I_1, I_2) and keypoints $\{j^p\}_{1:N_p}$ for persons $p \in \{1, \ldots, P\}$ in the image I_1 as input. During training, I_2 is generated by applying a randomly sampled affine warp to I_1. In this way, we do not need any annotated correspondences during training or pairs of images, but train the network on single images with annotated poses. We use an image resolution of 256×256 for both images.

The Siamese network generates features $F_1 \in \mathbb{R}^{32 \times 64 \times 64}$ and $F_2 \in \mathbb{R}^{32 \times 64 \times 64}$ for images I_1 and I_2, respectively. The features are then pixel-wise l_2 normalized and local descriptors $d_j^p \in R^{32 \times 3 \times 3}$ are generated for each keypoint j^p by extracting squared patches around the spatial position of a keypoint in the feature maps F_1.

Given a local descriptor d_j^p, we compute its affinity map A_j^p over all pixels $x = \{1, \ldots, 64\}$ and $y = \{1, \ldots, 64\}$ in F_2 as:

$$A_j^p = d_j^p \circledast F_2 \tag{1}$$

where $\circledast$ denotes the convolution operation. Finally, a softmax operation is applied to the affinity map A_j^p, *i.e.*,

$$S_j^p(x, y) = \frac{\exp(A_j^p(x, y))}{\sum_{x', y'} \exp(A_j^p(x', y'))}. \tag{2}$$

We refine the affinity maps S_j^p further using a refinement module.

4.2   Refinement Module

Similar to related multi-stage approaches [7, 10, 23], we append a second module to the keypoint correspondence network to improve the affinity maps generated by the Siamese matching module. For the refinement module, we use a batch normalized GoogleNet from layer 3 till layer 17. The refinement module concatenates F_1, F_2, and the affinity maps $\{S_j^p\}_{1:N_p}$ for a single person p and refines the affinity maps, which we denote by $C_j^p \in \mathbb{R}^{64 \times 64}$. The refinement module is therefore applied to the affinity maps for all persons $p \in \{1, \ldots, P\}$. Before we describe in Section 5 how we will use the affinity maps for tracking C_j^p, we describe how the keypoint correspondence network is trained.

4.3   Training

Since we train our network using self-supervision, we train it using a single image I_1 with annotated poses. We generate a second image I_2 by applying a randomly sampled affine warp to I_1. We then generate the ground-truth affinity map G_j^p for a keypoint j^p belonging to person p as:

$$G_j^p(x, y) = \begin{cases} 1 & \text{if } x = \hat{x}_j^p \text{ and } y = \hat{y}_j^p, \\ 0 & \text{otherwise,} \end{cases} \tag{3}$$

where $(\hat{x}_j^p, \hat{y}_j^p)$ is the spatial position of the ground-truth correspondence for keypoint j^p in image I_2, which we know from the affine transformation. As illustrated in Figure 3, not all corresponding keypoints are present in image I_2. In this case, the ground-truth affinity map is zero and predicting a corresponding keypoint is therefore penalized.

During training, we minimize the binary cross entropy loss between the predicted affinity maps S_j^p and C_j^p and the ground-truth affinity map G_j^p:

$$\min_{\theta} \sum_{x,y} - \left(G_j^p \log(S_j^p) + (1 - G_j^p)(1 - \log(S_j^p)) \right), \tag{4}$$

$$\min_{\theta} \sum_{x,y} - \left(G_j^p \log(C_j^p) + (1 - G_j^p)(1 - \log(C_j^p)) \right), \tag{5}$$

where θ are the parameters of the keypoint correspondence framework.

5   Multi-Person Pose Tracking

We use the keypoint correspondence network in two ways. First, we use it to recover human poses that have been missed by the frame-wise top-down multi-person pose estimation step, which will be described in Section 5.1. Second, we use keypoint correspondences for tracking poses across frames of the video as described in Section 5.2.
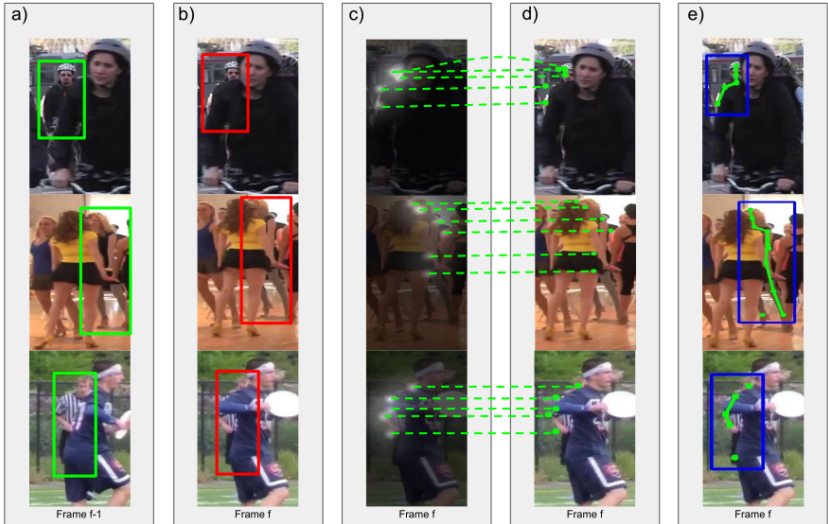
Fig. 4. Recovering missed detections. (a) Person detected by the top-down pose estimation framework in frame $f-1$. (b) Person missed by the top down pose estimation framework in frame f due to occlusion. (c) Keypoint affinity maps of the missed person from frame $f-1$ to frame f. (d) Corresponding keypoints in frame f. (e) Estimated bounding box from the corresponding keypoints and the recovered pose.

5.1   Recover Missed Detections

For a given frame f, we first detect the human poses in the frame using the top-down multi-person pose estimator described in Section 6.1. While the person detector [6] performs well, it fails in situations with overlapping persons and motion blur. Consequently, the human pose is not estimated in these cases. Examples are shown in Figure 4(b).

Given the detected human poses $J_{f-1}^p = \{j_{f-1}^p\}$ for persons $p \in \{1, \ldots, P\}$ in frame $f-1$, we compute the corresponding refined affinity maps $C^p = \{C_j^p\}$ by using the keypoint correspondence network. For each keypoint j_{f-1}^p, we then get the corresponding keypoint $\bar{j}_f^p$ in frame f by taking the argmax of C_j^p and mapping it to the image resolution. Since the resolution of the affinity maps is lower than the image resolution and since the frame f might contain a keypoint that was occluded in the previous frame, we reestimate the propagated poses. This is done by computing for each person p a bounding box that encloses all keypoints $\bar{J}_f^p = \{\bar{j}_f^p\}$ and using the human pose estimation network described in Section 6.1 to get a new pose for this bounding box. We denote the newly estimated poses by $\hat{J}_f^p$. The overall procedure is shown in Figure 4. We apply OKS based non-maximum suppression [42] to discard redundant poses.
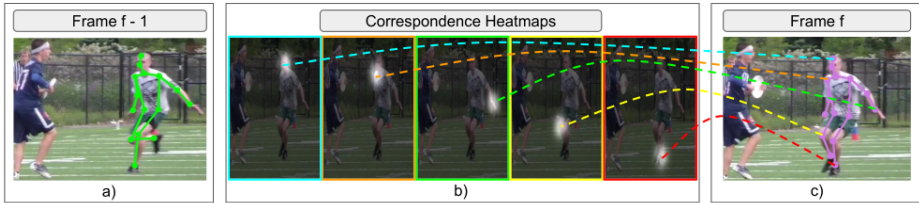
Fig. 5. Pose to track association. (a) A tracked human pose till frame $f-1$. (b) Keypoint affinity maps of the track to frame f. (c) A pose instance of frame f. The dashed lines indicate the position of each detected joint of the pose instance in the correspondence affinity maps of the tracked pose in frame $f-1$.

5.2 Tracking

Given detected and recovered poses, we need to link them across video frames to obtain tracks of human poses. Tracking can be seen as a data association problem over estimated poses. Previously, the problem has been approached using bipartite graph matching [13] or greedy approaches [42, 38, 11]. In this work, we greedily associate estimated poses over time by using the keypoint correspondences. We initialize tracks on the first frame and then associate new candidate poses to intial tracks one frame at a time.

Formally, our goal is to assign pose instances $\{B_f^p\} = \{J_f^p\} \cup \{\hat{J}_f^p\}$ in frame f for persons $p \in \{1, \dots P\}$ to tracks $\{T_{f-1}^q\}$ till frame $f-1$ for persons $q \in \{1, \dots Q\}$. Towards this end, we measure the similarity between a pose instance B_f^p and a track T_{f-1}^q as:

$$S(T_{f-1}^q, B_f^p) = \frac{\sum_{j=1}^{N_q} C_j^q(j_f^p) \cdot \mathbb{I}_{C_j^q(j_f^p) > \tau_{corr}}}{\sum_{j=1}^{N_q} \mathbb{I}_{C_j^q(j_f^p) > \tau_{corr}}}, \tag{6}$$

where C_j^q is the affinity map of the keypoint j in track T_{f-1}^q for frame f. The affinity map is computed by the network described in Section 4. $C_j^q(j_f^p)$ is the confidence value in the affinity map C_j^q at the location of the joint j_f^p for person p in frame f. N_q is the number of detected joints. An example is shown in Figure 5. We only consider j_f^p if its affinity is above τ_{corr}. If a pose B_f^p cannot be matched to a track T_{f-1}^q, a new track is initiated.

6 Experiments and Results

We evaluate our approach on the Posetrack 2017 and 2018 datasets [1]. The datasets have 292 and 593 videos for training and 214 and 375 videos for evaluation, respectively. We evaluate multi-frame pose estimation and tracking results using the mAP and MOTA evaluation metrics.

6.1   Implementation Details

We provide additional implementation details for our top-down pose estimation and keypoint correspondence network below.

Top-down Pose Estimation. We use a top-down framework for frame level pose estimation. We use cascade R-CNN [6] for person detection and extract crops of size $384{\times}288$ around detected persons as input to our pose estimation framework, which consists of two stages. Each stage is a batch normalized GoogleNet [39]. The backbone in the first stage consists of layer 1 to layer 17 while the second backbone consists of layer 3 to layer 17 only. Both stages predict pose heatmaps and joint offset maps for the cropped person as in [46]. We use the pose heatmaps in combination with the joint offsets from the second stage as our pose detections. The number of parameters (39.5 M) of our model is significantly lower compared to related works such as FlowTrack [42] (63.6 M) or EOGN [46] (60.3 M).

We train the pose estimation framework on the MS-COCO dataset [27] for 260 epochs with a base learning rate of $1e^{-3}$. The learning rate is reduced to $1e^{-4}$ after 200 epochs. During training we apply random flippings and rotations to input crops. We finetune the pose estimation framework on the PoseTrack 2017 dataset [1] for 12 epochs. The learning rate is further reduced to $1e^{-5}$ after epoch 7.

Keypoint Correspondence Network. We perform module-wise training. We first train the Siamese module. We then fix the Siamese module and train the refinement module. Both modules are trained for 100 epochs with base learning rate of $1e^{-4}$ reduced to $1e^{-5}$ after 50 epochs. We generate a second image for each training image by applying random translations, rotations, and flippings to the first image. The keypoint correspondence network is trained only on the MS-COCO dataset [27]. We did not observe any improvements in our tracking results by finetuning the correspondence model on the PoseTrack dataset. Training only on the PoseTrack dataset yielded sub-optimal tracking results since PoseTrack is sparsely annotated and contains far less person instances than MS-COCO.

6.2   Baselines

We compare our keypoint correspondence tracking to different standard tracking baselines for multi-person pose tracking as reported in Table 2. To measure the performance of each baseline, we report the number of identity switches and the MOTA score. For a fair comparison, we replace the keypoint correspondences in our framework by different baselines. For all experiments, we use the same detected poses using either ground truth or detected bounding boxes.

OKS. OKS without taking the motion of the poses into account has been proposed in [42]. OKS measures the similarity between two poses and is independent of their appearance. It is not robust to large motion, occlusion, and large temporal offsets. This is reflected in Table 2 as this baseline achieves the lowest performance.

Table 2. Comparison with tracking baselines on the PoseTrack 2017 validation set. For the comparison, the detected poses based on ground-truth bounding boxes (GT Boxes) or detected bounding boxes are the same for each approach. Correspondence based tracking consistently improves MOTA compared to the baselines and significantly reduces the number of identity switches (IDSW).

Tracking Method	GT Boxes	IDSW	MOTA
OKS	✓	6582	65.9
Optical Flow	✓	4419	68.4
Re-ID	✓	4164	67.1
Correspondences	✓	3583	70.5
OKS	✗	7207	60.4
Optical Flow	✗	5611	66.7
Re-ID	✗	4589	64.1
Correspondences	✗	3632	67.9

Optical Flow. Optical flow is a temporal baseline that has been proposed in [42]. We use optical flow to warp the poses from the previous frame to the current frame. We then apply OKS for associating the warped poses with candidate poses in the current frame. We use the pre-trained PWC-net [37] as done in [37] for a fair comparison. Optical flow clearly outperforms OKS and achieves superior MOTA of 68.4 and 66.7 for GT and detected bounding boxes, respectively.

Person Re-id. Compared to optical flow and OKS, person re-identification is more robust to larger temporal offsets and large motion. However, the achieved results indicate that person re-identification operating on the bounding boxes performs sub-optimally under the frequent partial occlusions in the PoseTrack datasets. For our experiments, we use the pre-trained re-identification model from [28]. Re-identification based tracking achieves MOTA scores of 67.1 and 64.1 for GT and detected bounding boxes, respectively.

The results show that correspondence based tracking (1) achieves a consistent improvement over the baselines for ground-truth and detected bounding boxes with MOTA scores of 70.5 and 67.9, respectively, and (2) significantly reduces the number of identity switches. Compared to optical flow, correspondences are more robust to partial occlusions, motion blur, and large motions. A qualitative comparison is provided in Section 6.5.

6.3 Effect of Joint Detection Threshold and Pose Recovery

We evaluate the impact of different joint detection thresholds on mAP and MOTA for the PoseTrack 2018 dataset as shown in Table 3. Since mAP does not penalize false-positive keypoints, thresholding decreases the pose estimation performance by discarding low confident joints. Vice versa, joint thresholding results in cleaner tracks and improves the tracking performance, as MOTA penalizes false-positive keypoint detections. A good trade-off between mAP and MOTA is achieved for the joint detection threshold 0.3 resulting in mAP and MOTA of 77.7 and 67.6, respectively.

Table 3. Effect of joint detection threshold and pose recovery on mAP and MOTA for the PoseTrack 2018 validation set. The results are shown for (left) detected poses only and (right) detected and recovered poses. As expected, recovering missed detections improves both MOTA and mAP. A good trade-off between mAP and MOTA is achieved by the joint detection threshold 0.3.

Joint Threshold	mAP	MOTA		Joint Threshold	mAP	MOTA
Detected Poses Only				Detected and Recovered Poses.		
0.0	80.1	48.1		0	82.0	48.1
0.1	79.7	63.3		0.1	81.4	64.1
0.2	78.9	66.1		0.2	80.5	67.2
0.3	**77.7**	**67.6**		**0.3**	**79.2**	**68.8**
0.4	75.9	68.0		0.4	77.2	69.2
0.5	73.1	67.1		0.5	74.2	68.2

Table 4. Comparison to the state-of-the-art on the PoseTrack 2017 and 2018 validation set for multi-frame pose estimation.

Dataset	Method	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	mAP
PoseTrack 17 Val Set	DetectNTrack [13]	72.8	75.6	65.3	54.3	63.5	60.9	51.8	64.1
	PoseFlow [43]	66.7	73.3	68.3	61.1	67.5	67.0	61.3	66.5
	FlowTrack [42]	81.7	83.4	80.0	72.4	75.3	74.8	67.1	76.7
	HRNet [38]	82.1	83.6	80.4	73.3	75.5	75.3	68.5	77.3
	MDPN [14]	85.2	88.5	83.9	78.0	82.4	80.5	73.6	80.7
	PoseWarper [4]	81.4	88.3	83.9	78.0	82.4	80.5	73.6	**81.2**
	Ours	86.1	87.0	83.4	76.4	77.3	79.2	73.3	80.8
PoseTrack 18 Val Set	PoseFlow [43]	63.9	78.7	77.4	71.0	73.7	73.0	69.7	71.9
	MDPN [14]	75.4	81.2	79.0	74.1	72.4	73.0	69.9	75.0
	PoseWarper [4]	79.9	86.3	82.4	77.5	79.8	78.8	73.2	79.7
	Ours	86.0	87.3	84.8	78.3	79.1	81.1	75.6	**82.0**

While on the left hand side of Table 3 we report the results without recovering missed detections as described in Section 5.1, the table on the right hand side shows the impact on mAP and MOTA if missed detections are recovered. The recovering of missed detections improves the accuracy for all thresholds. For the joint detection threshold 0.3, mAP and MOTA are further improved to 79.2 and 68.8, respectively.

6.4   Comparison with State-of-the-art Methods

We compare to the state-of-the-art for multi-frame pose estimation and multi-person pose tracking on the PoseTrack 2017 and 2018 datasets.

Multi-Frame Pose Estimation. For the task of multi-frame pose estimation, we compare to the state-of-the-art on the PoseTrack 2017 and 2018 validation sets, respectively. Although our correspondences are trained without using any video data, our approach outperforms the recently proposed PoseWrapper [4] approach on the PoseTrack 2018 validation set with mAP of 82.0 and achieves very competitive mAP on the PoseTrack 2017 validation set with mAP of 80.8 as shown in Table 4.

Table 5. Comparison to the state-of-the-art on the PoseTrack 2017 and 2018 valida-tion and test sets. Approaches marked with [+] use additional external training data. Approaches marked with [*] do not report results on the official test set

	Approach	mAP	MOTA		Approach	mAP	MOTA
PoseTrack 17 val set	STEmbedding [21][*]	77.0	71.8	PoseTrack 18 val set	Ours + Merge	79.2	69.1
	EOGN [46]	76.7	70.1		Ours	79.2	68.8
	PGPT [3]	77.2	68.4		MIPAL [18]	74.6	65.7
	Ours + Merge	**78.0**	**68.3**		LightTrack [33]	71.2	64.9
	Ours	**78.0**	**67.9**		Miracle[+] [45]	80.9	64.0
	POINet [36]	-	65.9		OpenSVAI [34]	69.7	62.4
	HRNet [38]	77.3	-		STAF [35]	70.4	60.9
	FlowTrack [42]	76.7	65.4				
PoseTrack 17 test set	EOGN [46]	74.8	61.1	PoseTrack 18 test set	MSRA[+]	74.0	61.4
	PGPT [3]	72.6	60.2		ALG[+]	74.9	60.8
	Ours + Merge	**74.2**	**60.0**		**Ours + Merge**	**74.4**	**60.7**
	POINet [36]	72.5	58.4		Miracle[+] [45]	70.9	57.4
	LightTrack [33]	66.8	58.0		MIPAL [18]	67.8	54.9
	HRNet [38]	75.0	58.0		CV-Human	64.7	54.5
	FlowTrack	74.6	57.8				

Multi-Person Pose Tracking. We compare our tracking approach with the state-of-the-art for multi-person pose tracking on the PoseTrack 2017 and 2018 val-idation sets and leaderboards. In addition, we perform a post-processing step in which we merge broken tracks similar to the recovery of missed detections described in Section 5.1. This further improves the tracking performance. For further details we refer to the supplementary material.

We submitted our results to the PoseTrack 2017 and 2018 test servers, respec-tively. Our approach achieves top scoring MOTA of 60.0 on the PoseTrack 2017 leaderboard without any bells and whistles as shown in Table 5. Our tracking performance is on-par with state-of-the-art approaches on the PoseTrack 2017 validation set.

Similarly, we achieve top scoring MOTA of 69.1 on the PoseTrack 2018 vali-dation set as shown in Table 5. Our tracking results are very competitive to the winning entries on the PoseTrack 2018 leaderboard although the winning entries use additional training data.

6.5   Qualitative Results

We qualitatively compare optical flow and correspondences for the task of pose warping under motion blur, occlusions, and large motion in Figure 6. While the column on the left hand side shows the query pose in frame f. The columns in the middle and on the right hand side show the warped poses generated by optical flow or correspondences, respectively. In contrast to optical flow, our approach is robust to occlusion and fast human or camera motion. Our approach, however, has also some limitations. For instance, we observe that we obtain sometimes two tracks for the same person if the person detector provides two or more bounding boxes for a person like one bounding box for the upper body and one bounding box for the full body. Examples of failure cases are shown in the supplementary material.
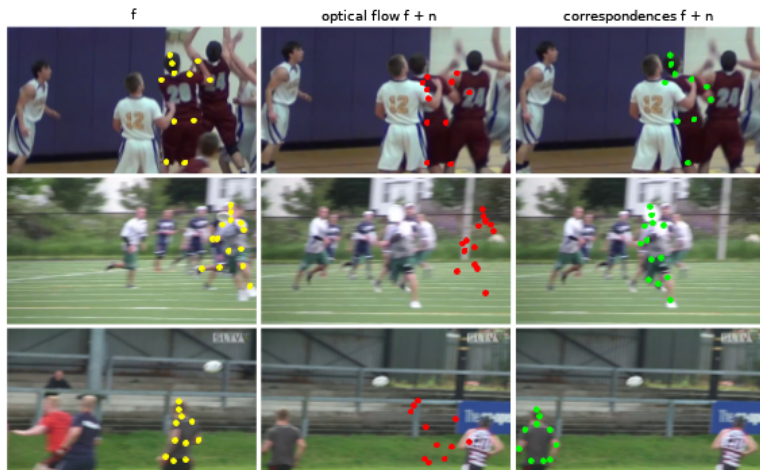
Fig. 6. Qualitative comparison between optical flow and correspondences for the task of pose warping under occlusion, motion blur, and large motion. (a) Query pose in frame f. (b) Warped pose using optical flow. (c) Warped pose using correspondences. In contrast to optical flow, the correspondences warp the poses correctly despite of occlusions, motion blur, or large motion.

7   Conclusion

In this work, we have proposed a self-supervised keypoint correspondence framework for the tasks of multi-frame pose estimation and multi-person pose tracking. The proposed keypoint correspondence framework solves two tasks: (1) recovering missed detections and (2) associating human poses across video frames for the task of multi-person pose tracking. The proposed approach based on keypoint correspondences outperforms the state-of-the-art for the tasks of multi-frame pose estimation and multi-person pose tracking on the PoseTrack 2017 and 2018 datasets.

References

1. Andriluka, M., Iqbal, U., Ensafutdinov, E., Pishchulin, L., Milan, A., Gall, J., B., S.: PoseTrack: A benchmark for human pose estimation and tracking. In: CVPR (2018)

2. Andriluka, M., Pishchulin, L., Gehler, P., Schiele, B.: 2d human pose estimation: New benchmark and state of the art analysis. In: CVPR (2014)

3. Bao, Q., Liu, W., Cheng, Y., Zhou, B., Mei, T.: Pose-guided tracking-by-detection: Robust multi-person pose tracking. IEEE Transactions on Multimedia (2020)

4. Bertasius, G., Feichtenhofer, C., Tran, D., Shi, J., Torresani, L.: Learning temporal pose estimation from sparsely-labeled videos. In: NeurIPS (2019)

5. Bertinetto, L., Valmadre, J., Henriques, J.F., Vedaldi, A., Torr, P.H.: Fully-convolutional siamese networks for object tracking. In: ECCV (2016)

6. Cai, Z., Vasconcelos, N.: Cascade R-CNN: Delving into high quality object detection. In: CVPR (2017)

7. Cao, Z., Simon, T., Wei, S.E., Sheikh, Y.: Realtime multi-person 2d pose estimation using part affinity fields. In: CVPR (2017)

8. Chen, Y., Wang, Z., Peng, Y., Zhang, Z., Yu, G., Sun, J.: Cascaded pyramid network for multi-person pose estimation. In: CVPR (2018)

9. Choy, C., Gwak, J., Savarese, S., Chandraker, M.: Universal correspondence network. In: NIPS (2016)

10. Dantone, M., Gall, J., Leistner, C., van Gool, L.: Body parts dependent joint regressors for human pose estimation in still images. TPAMI (2014)

11. Doering, A., Iqbal, U., Gall, J.: Joint flow: Temporal flow fields for multi person tracking. In: BMVC (2018)

12. Felzenszwalb, P.F., Huttenlocher, D.P.: Pictorial structures for object recognition. IJCV (2005)

13. Girdhar, R., Gkioxari, G., Torresani, L., Paluri, M., Tran, D.: Detect-and-Track: Efficient Pose Estimation in Videos. In: CVPR (2018)

14. Guo, H., Tang, T., Luo, G., Chen, R., Lu, Y., Wen, L.: Multi-domain pose network for multi-person pose estimation and tracking. In: CVPR (2018)

15. Han, K., Rezende, R., Ham, B., Wong, K.Y., Cho, M., Schmid, C., Ponce., J.: Scnet: Learning semantic correspondence. In: ICCV (2017)

16. He, K., Gkioxari, G., Dollár, P., Girshick, R.B.: Mask R-CNN. ICCV (2017)

17. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)

18. Hwang, J., Lee, J., Park, S., Kwak, N.: Pose estimator and tracker using temporal flow maps for limbs. In: IJCNN (2019)

19. Insafutdinov, E., Andriluka, M., Pishchulin, L., Tang, S., Levinkov, E., Andres, B., Schiele, B.: ArtTrack: Articulated Multi-person Tracking in the Wild. In: CVPR (2017)

20. Iqbal, U., Milan, A., Gall, J.: Posetrack: Joint multi-person pose estimation and tracking. In: CVPR (2017)

21. Jin, S., Liu, W., Ouyang, W., Qian, C.: Multi-person articulated tracking with spatial and temporal embeddings. In: CVPR (2019)

22. Kim, S., Min, D., Ham, B., Jeon, S., Lin, S., Sohn, K.: Fully convolutional self-similarity for dense semantic correspondence. In: CVPR (2017)

23. Kocabas, M., Karagoz, S., Akbas, E.: MultiPoseNet: Fast multi-person pose estimation using pose residual network. In: ECCV (2018)

24. Li, B., Yan, J., Wu, W., Zhu, Z., Hu, X.: High performance visual tracking with siamese region proposal network. In: CVPR (2018)

25. Li, W., Wang, Z., Yin, B., Peng, Q., Du, Y., Xiao, T., Yu, G., Lu, H., Wei, Y., Sun, J.: Rethinking on multi-stage networks for human pose estimation. arXiv preprint (2019)

26. Li, X., Liu, S., Mello, S.D., Wang, X., Kautz, J., Yang, M.H.: Joint-task self-supervised learning for temporal correspondence. In: NeurIPS (2019)

27. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.: Microsoft COCO: Common objects in context. In: ECCV (2014)
28. Luo, H., Gu, Y., Liao, X., Lai, S., Jiang, W.: Bag of tricks and a strong baseline for deep person re-identification. In: CVPR Workshop (2019)
29. Moon, G., Chang, J.Y., Lee, K.M.: Multi-scale aggregation R-CNN for 2d multi-person pose estimation. In: CVPR Workshop (2019)
30. Newell, A., Huang, Z., Deng, J.: Associative embedding: End-to-end learning for joint detection and grouping. In: NIPS (2017)
31. Newell, A., Yang, K., Deng, J.: Stacked hourglass networks for human pose estimation. In: ECCV (2016)
32. Nie, X., Feng, J., Xing, J., Yan, S.: Generative partition networks for multi-person pose estimation. In: ECCV (2018)
33. Ning, G., Huang, H.: Lighttrack: A generic framework for online top-down human pose tracking. arXiv preprint (2019)
34. Ning, G., Liu, P., Fan, X., Zhang, C.: A top-down approach to articulated human pose estimation and tracking. In: ECCV Workshop (2019)
35. Raaj, Y., Idrees, H., Hidalgo, G., Sheikh, Y.: Efficient online multi-person 2d pose tracking with recurrent spatio-temporal affinity fields. In: CVPR (2019)
36. Ruan, W., Liu, W., Bao, Q., Chen, J., Cheng, Y., Mei, T.: POINet: Pose-guided ovonic insight network for multi-person pose tracking. In: International Conference on Multimedia (2019)
37. Sun, D., Yang, X., Liu, M.Y., Kautz, J.: PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume. In: CVPR (2018)
38. Sun, K., Xiao, B., Liu, D., Wang, J.: Deep high-resolution representation learning for human pose estimation. In: CVPR (2019)
39. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: CVPR (2015)
40. Wang, X., Jabri, A., Efros, A.A.: Learning correspondence from the cycle-consistency of time. In: CVPR (2019)
41. Wu, J., Zheng, H., Zhao, B., Li, Y., Yan, B., Liang, R., Wang, W., Zhou, S., Lin, G., Fu, Y., Wang, Y., Wang, Y.: AI challenger: A large-scale dataset for going deeper in image understanding. arXiv preprint (2017)
42. Xiao, B., Wu, H., Wei, Y.: Simple baselines for human pose estimation and tracking. In: ECCV (2018)
43. Xiu, Y., Li, J., Wang, H., Fang, Y., Lu, C.: Pose Flow: Efficient online pose tracking. In: BMVC (2018)
44. Yang, W., Li, S., Ouyang, W., Li, H., Wang, X.: Learning feature pyramids for human pose estimation. In: ICCV (2017)
45. Yu, D., Su, K., Sun, J., Wang, C.: Multi-person pose estimation for pose tracking with enhanced cascaded pyramid network. In: ECCV Workshop (2018)
46. Zhang, R., Zhu, Z., Li, P., Wu, R., Guo, C., Huang, G., Xia, H.: Exploiting offset-guided network for pose estimation and tracking. In: CVPR (2019)
47. Zhang, Z., Peng, H., Wang, Q.: Deeper and wider Siamese networks for real-time visual tracking. In: CVPR (2019)
48. Zhu, Z., Wang, Q., Li, B., Wu, W., Yan, J., Hu, W.: Distractor-aware siamese networks for visual object tracking. In: ECCV (2018)