A Gated Attention Transformer for Multi-Person Pose Tracking

Andreas Doering[1,2] Juergen Gall[1,2]
[1]University of Bonn [2]Lamarr Institute for Machine Learning and Artificial Intelligence

Abstract

Multi-person pose tracking is an important element for many applications and requires to estimate the human poses of all persons in a video and to track them over time. The association of poses across frames remains an open research problem, in particular for online tracking methods, due to motion blur, crowded scenes and occlusions. To tackle the association challenge, we propose a Gated Attention Transformer. The core aspect of our model is the gating mechanism that automatically adapts the impact of appearance embeddings and embeddings based on temporal pose similarity in the attention layers. In order to re-identify persons that have been occluded, we incorporate a pose-conditioned re-identification network that provides initial embeddings and allows to match persons even if the number of visible joints differ between frames. We further propose a matching layer based on gated attention for pose-to-track association and duplicate removal. We evaluate our approach on PoseTrack 2018 and PoseTrack21.

1. Introduction

Multi-person pose tracking is highly relevant for a wide range of applications such as virtual reality, autonomous driving or sports analysis and requires to accurately estimate and track the human poses of all persons throughout a video. Despite of the recent progress in multi-person pose tracking [28, 11, 19, 13, 32, 41, 29, 42, 37], the task remains very challenging due to camera motion, motion blur, occlusions, and a high variety in pose and scale [10]. Consequently, a tracking approach must be robust to detection errors and ambiguities. In particular, the assignment of highly occluded persons in unusual poses is very difficult as shown in Fig. 1, where three persons perform a gymnastic exercise in water. For instance, the green bounding box intersects with three persons and the respective pose overlaps with the keypoints of the other persons. This poses a challenge, especially for on-line methods as assignments will be made once a new frame arrives.

Related works such as [32, 29, 42, 41, 47] try to tackle these challenges by generating future poses from a track's history, which are then matched with detections based on pose similarities, *e.g.*, based on Object Keypoint Similarity (OKS) [29, 41, 47] or a pose-based matching layer [32]. Other works such as [37] process each sequence in an offline fashion, which is not feasible for real-time applications. As these works mainly rely on pose-based similarities for matching, these methods tend to fail to re-identify tracks that have been occluded for longer periods of time or undergo high pose deformations.

In our work, we thus focus on learning the association of detected persons to tracks in an on-line fashion and propose an approach that leverages the estimated poses, bounding boxes, and the appearance of the detected persons to assign them to previous tracks or initialize new tracks. Since we can neither rely solely on appearance-based features nor pose similarities due to multiple instances with similar appearance, changing camera views or scene switches, which often occur in in-the-wild sequences, we introduce two types of embeddings. The detection and track embeddings are based on appearance and used to measure the appearance similarity between detections and previous tracks. The additional edge embeddings directly encode the pose similarity between a pose and a track based on estimated poses and bounding boxes. While the pose similarity is a strong prior for tracking, it fails in case of fast motion or if the person disappeared for some frames due to occlusion or being outside of the camera view. We thus propose a gated attention transformer that combines and weights the attention matrices of both embedding types. All three embeddings are updated by a gated attention decoder and a final matching layer assigns the detections to the tracks. The matching layer also removes duplicates, *i.e.*, multiple detections for the same persons, and initiates new tracks. Furthermore, we employ a pose-conditioned re-identification model where the appearance features are normalized based on the heatmaps of the detected keypoints.

We evaluate our approach on the challenging PoseTrack 2018 [1] and PoseTrack21 [10] datasets where it achieves state-of-the-art results. In summary, we propose i) a gated attention transformer that combines pose and appearance similarity in a novel way for pose-to-track association and ii) a novel matching layer for pose-to-track association and

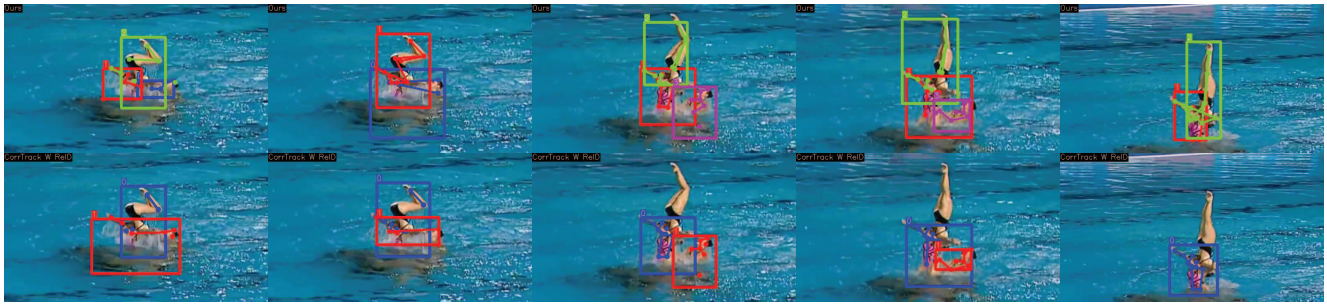

Figure 1. Qualitative examples of our proposed method on the PoseTrack21 dataset. The first row contains visual tracking results of our method and the second row shows visualizations of CorrTrack with ReID [10].

duplicate removal.

2. Related Work

We briefly discuss some related works for person re-identification and multi-person pose tracking.

Person Re-Identification: Methods for person re-identification aim to generate robust feature representations of a query person, which allows to re-identify each instance of the particular person. These methods can be divided into different categories such as re-identification based on global features [25, 18, 8, 40], part-based approaches [34, 22, 33], prior-based, *i.e*., pose- or mask-guided [14, 30, 46, 39, 27] and video-based [6, 5, 23] re-identification. Part-based models [34, 22, 33] divide each image into several parts and extract distinct part-based features. Features for each part are either trained individually [34] or further re-combined to obtain visibility-aware features [33]. Li *et al*. [22] propose to learn discriminative implicit parts (DiP) based on a vision transformer (ViT) [12] architecture, which tokenizes each image into equally sized patches. Pose- [14, 30, 46, 39] or mask-guided [27] approaches aim to suppress background-noise by learning occlusion- and instance-aware features that are more robust in crowded scenarios with occlusions. In this work, we propose Spatially Adaptive Pose DEnormalization (SPAPDE) layers for pose-guided re-identification and use it in the proposed gated attention transformer for multi-person pose tracking.

Multi-Person Pose Tracking: Existing works on multi-person pose tracking can be divided into two categories: top-down methods [42, 37, 32, 29, 10, 47, 43] and bottom-up methods [28, 19, 13, 11]. Former approaches employ a person detector and estimate the pose for every detected person individually based on temporal-context. Most methods employ a pose-warping [29, 10, 43] scheme that warps tracked poses into the next frame or directly predict poses based on the tracklet history [42, 32, 47, 21, 11], which are then matched with detected poses using greedy or Hungarian matching. In [37], an offline approach has been proposed that merges multiple overlapping fixed-lengths tracklets into tracks based on bipartite matching and Dijkstra's algorithm [9]. Bottom-up approaches [28, 19, 21, 11], on the other hand, predict all keypoints within an image simultaneously and generate tracks by solving spatio-temporal graphs between detected keypoints. For instance, [28, 11, 19] generate spatio-temporal vector fields, while various spatio-temporal embeddings for the association of keypoints and tracks are proposed in [21]. In contrast, person instances are tracked in [13] using a semi-supervised approach based on video instance correspondences. In this work, we propose a gating mechanism that automatically adapts the impact of appearance embeddings and edge embeddings, which are a strong prior and encode only pose and bounding box similarity, in the attention layers.

3. Gated Attention Transformer for Multi-Person Pose Tracking

On-line methods for multi-person pose tracking often follow the tracking-by-detection paradigm [32, 15, 11, 28, 19, 41, 44, 29] and usually suffer from ambiguities and occlusions. An example is shown in the second row of Fig. 1 where the blue id jumps between two persons. In order to make the matching between detections and previously tracked persons more robust, we propose a gated attention transformer that directly learns the matching by a gated matching layer. It combines appearance features and encoded temporal person similarities. Since the importance of appearance and pose similarity varies within a video and between videos, in particular when a person has been occluded for a few frames, the gated attention decoder and the matching layer use a gating mechanism to update the embeddings of the new poses and previous tracks and to match poses to tracks. For example, if there are several very similar looking persons in a frame, the pose similarity can guide the update of the appearance embedding. Vice versa, the update will be driven by the appearance embeddings if there is no spatial proximity between tracks and detections. Our approach for multi-person pose tracking is illustrated in Fig. 2.

We assume that the human poses are extracted for a new frame t by a standard multi-person pose estimator where we utilize the detector from [10] for a fair comparison. Specifi-

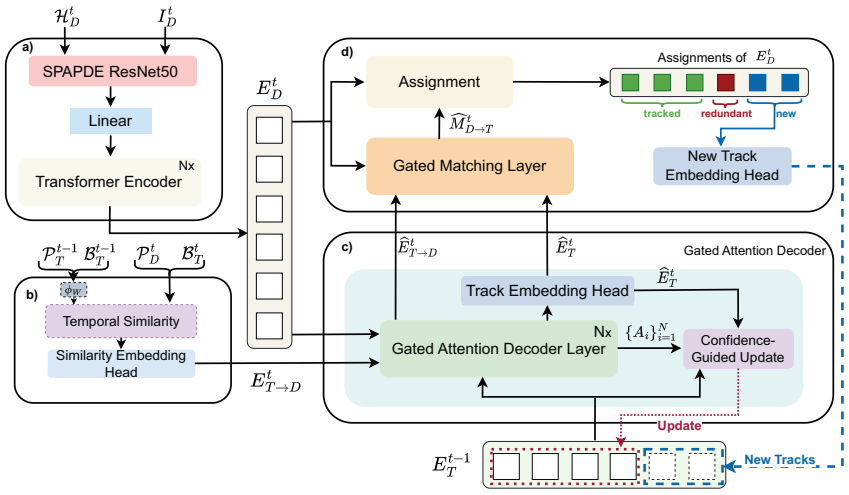

Figure 2. The proposed multi-person pose tracking architecture entails the following steps: a) Given a set of person crops and their respective keypoint heatmaps at time frame t, we compute pose-conditioned appearance features and feed them into N transformer encoder stages. This gives an embedding for each detection E_D^t, which will be used to measure the similarity to previous tracks. b) We also compute spatial similarities between tracks and detected persons by means of Intersection over Union (IoU) and Object Keypoint Similarity (OKS), which we then encode into a pose similarity embedding $E_{T\to D}^t$. This serves as a strong prior for matching. c) The gated attention decoder takes the embeddings of the previous tracks E_T^{t-1}, the detection embeddings E_D^t and the pose similarity embeddings $E_{T\to D}^t$ as input and updates the embeddings. It adaptively weights the spatial prior $E_{T\to D}^t$ and the appearance similarity between E_T^{t-1} and E_D^t. d) Finally, the matching stage assigns detections to tracks, removes redundant detections and initializes new tracks.

cally, for a given frame at time t, our network takes as input the set of estimated bounding boxes $\mathcal{B}_D^t$ and poses $\mathcal{P}_D^t$, and additionally detected keypoint heatmaps $\mathcal{H}_D^t$ extracted from the image crops $\mathcal{I}_D^t$ of detected persons. From the heatmaps and image crops, an embedding E_D^t for re-identification is computed as described in Section 3.3 (Fig. 2a). To measure pose and spatial proximity between detections and tracks, similarities to the bounding boxes $\mathcal{B}_T^{t-1}$ and poses $\mathcal{P}_T^{t-1}$ of the last frame of each track are computed, which results in the edge embedding $E_{T\to D}^t$ (Fig. 2b) between a track T and a detection D, which will be described in Section 3.3 as well. We denote $E_{T\to D}^t$ as pose similarity embedding. Given both embeddings E_D^t and $E_{T\to D}^t$ as well as the track embeddings E_T^{t-1} that have been estimated in the previous frame $t-1$, the proposed *Gated Attention Decoder* (Fig. 2c) and the *Gated Matching Layer* (Fig. 2d) assign the current detections to previous tracks, update the track embeddings, and initialize new tracks. Both will be described in the following Section 3.1.

3.1. Gated Attention Decoder

Attention Layer: As baseline method, we employ a transformer decoder as proposed in [36] and we propose an attention-based matching layer as the main extension. As illustrated in Fig. 3a, the attention layer calculates the similarities $A = \sigma(O_A)$ between the appearance features of the detections E_D^t and tracks E_T^{t-1}, where σ is the row-wise softmax layer. O_A represents the appearance-based similarity logits that are obtained by cross-attention between the appearance features of the detections E_D^t and tracks E_T^{t-1}:

$$O_A = \frac{E_T^{t-1} W_Q^\intercal \left(E_D^t W_K^\intercal\right)^\intercal}{\sqrt{d}}, \tag{1}$$

where W_Q and W_K are the learned projection weights for the queries and keys, respectively, and d is the dimensionality of the embeddings E_*.

In order to assign detections to tracks based on appearance similarity, we need to allow that none of the detections is assigned to a track, *e.g.*, if a person has not been detected or is occluded. Prior to applying the softmax to the attention logits, *i.e.*, $A = \sigma(O_A)$, we thus add a column of zeros. In other words, the last column of A indicates if a track does not match with any detection.

Finally, the attention layer calculates the proposed track embedding update as follows:

$$\Delta E_T^t = (A_{:,:-1} E_D^t) W_A^\intercal, \tag{2}$$

where $A_{:,:-1}$ denotes the attention weights without the last column and W_A are the weights of a linear layer.

Gated Attention Layer: Appearance similarities allow to re-identify an occluded person after some frames, but are unreliable in case of motion blur or person instances with similar appearance as it is common in team sport videos. Pose and spatial similarities, on the other hand, provide a

strong matching prior, but are less reliable in crowded scenarios as shown in Fig. 1. While none of them can resolve all ambiguities, fusing the similarities automatically provides a stronger matching prior. Based on these intuitions, we propose the gated attention layer. As illustrated in Fig. 3d, the gated attention layer extends Fig. 3a and incorporates pose similarity weights $S_E{=}\sigma\left(O_E\right)$ between detections and tracks. The pose similarity logits $O_E{=}E^t_{T\to D}W^\intercal_E$ are obtained by $E^t_{T\to D}$ and the learned weight matrix W_E.

In order to assign detections to tracks, appearance-based similarities S_A and pose-based similarities S_E are fused by the $\alpha{-}Gate$ that weights the contribution of appearance-based attention weights and the pose-based attention weights by a hyperparameter α, which we evaluate in our experiments:

$$A = \alpha \cdot S_A + (1-\alpha)\cdot S_E. \tag{3}$$

Similar to the attention layer (Fig. 3a), we add a column of zeros to S_A and S_E and obtain ΔE^t_T following (2). Intuitively, fusing the normalized similarities S_A and S_E automatically assigns a higher weight to the similarity measure where the matching confidence of a detection to a track is higher, resulting in a higher tracking accuracy, as we will show in the experiments.

Decoder Layer: The decoder layer shown in Fig. 3b then updates the track E^{t-1}_T embeddings based on the output of the attention layer. As common for transformer blocks [36], we use a residual feed-forward network (FNN) as shown in Fig. 3b. Specifically, we compute

$$\widehat{E}^t_T = LN(\widetilde{E}^t_T{+}FFN(\widetilde{E}^t_T)), \qquad \widetilde{E}^t_T = LN(E^{t-1}_T{+}\Delta E^t_T), \tag{4}$$

where LN denotes layer normalization.

Gated Decoder Layer: The gated decoder layer as shown in Fig. 3e) additionally employs an $\alpha{-}Gate$ to weight the contribution of appearance-based and pose similarity-based attention logits O_A and O_E similar to the gated attention layer. We then apply a feed-forward network FFN_E on the weighted sum of attention logits

$$\widehat{E}^t_{T\to D} = FFN_E\left(\alpha\cdot O_A + (1-\alpha)\cdot O_E\right), \tag{5}$$

to update the pose similarity embeddings, where FFN_E is a feed-forward network as FFN. Empirically, the gated decoder layer performs best if α-gating is performed on the attention logits as we show in the experiments.

Gated Matching Layer: The matching layer (Fig. 3c) and the gated matching layer (Fig. 3f) comprise a structure similar to the attention layer and the gated attention layer, respectively, but differ in two aspects. 1) Both matching layers (Fig. 3c) and (Fig. 3f) do not use the attention weights to predict a track embedding update and therefore

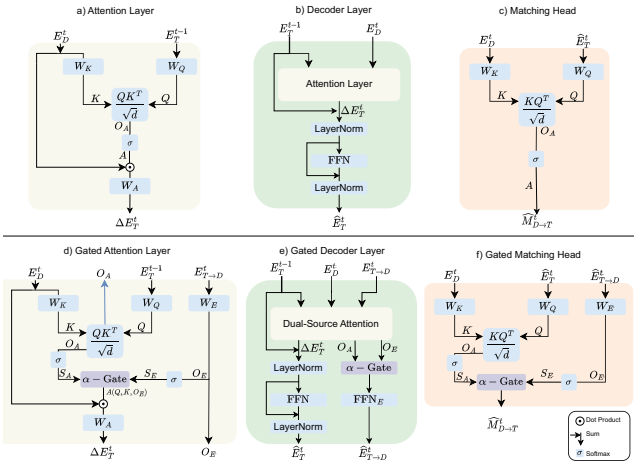

Figure 3. Illustration of our proposed d) Gated Attention layer, e) Gated Decoder layer and f) Gated Matching layer. The top row shows a vanilla implementation without gating for each layer: a) Standard Attention layer with E^t_D as keys and E^{t-1}_T as queries. b) The Decoder layer takes as input the detection embeddings E^t_D and track embeddings E^{t-1}_T and updates the track embeddings $\hat{E}^t_T$ using the Attention layer. c) The Matching layer predicts an assignment matrix $\widehat{M}^t_{D\to T}$ of detections to tracks. The proposed gated layers use the additional pose similarity embeddings $\hat{E}^t_{T\to D}$ as strong prior. The α-Gates fuse the cross-attention matrix S_A between detections and tracks, which measures appearance similarity, and the pose similarity matrix S_E, which measures spatial and pose similarity between detections and tracks.

do not consist of a linear layer after the softmax and α-Gate, respectively. 2) The attention weights A (*i.e.* (3)) are used as assignment matrix $\widehat{M}^t_{D\to T}$ to assign detections to tracks, as we show in Fig. 2d. In the presence of duplicate detections, this allows to match multiple detections to a single track. Given the matching matrix $\widehat{M}^t_{D\to T}$, we utilize Hungarian matching to assign detections to tracks. All the remaining detections i that have a matching probability $\hat{m}_{i\to j} > \tau_{dup}$ for any track j are considered as duplicate detections and are removed. Any other detection will initialize a new track embedding (Fig. 2d). All tracks that have not been tracked for τ_{age} frames will be removed. We evaluate the impact of τ_{dup} and τ_{age} in the experiments.

As shown in Fig. 2, we employ an additional *Track Embedding Head* after the last decoder layer that predicts the final track embeddings and *New Track Embedding Head* that generates the embedding for newly initialized tracks. Both heads consist of two linear layers, where the first layer includes a LayerNorm [2] and GELU [17].

3.2. Confidence-Guided Track Update

As shown in Fig. 2c, we perform the final update of the tracks embedding before the final assignment of detections to tracks is carried out by the matching layer (Fig. 2d). This

Approach	Online	AssA	FragA	DetA	HOTA	MOTA	mAP
CorrTrack [10]	✓	58.02	57.75	45.48	51.13	63.0	72.3
CorrTrack w. ReID [10]	✓	60.21	59.66	46.56	52.71	63.8	72.7
Tracktor++ w. Poses [10]	✓	59.41	58.61	46.30	52.21	63.3	71.4
Tracktor++ w. Corr. [10]	✓	54.05	52.02	44.67	48.90	61.6	**73.6**
Ours w/o gating	✓	44.92	41.96	45.30	44.82	52.4	70.2
Ours w. gating	✓	**62.20**	**60.93**	**47.20**	**53.94**	**64.1**	**73.6**
CorrTrack [10]	✗	60.93	60.37	45.48	52.42	63.9	72.3

Table 1. Comparison to multi-person pose tracking methods on the PoseTrack21 dataset.

Method	AssA	DetA	LocA	HOTA	MOTA
TRMOT [38]	54.98	40.91	79.92	46.85	47.2
FairMOT [45]	61.45	47.43	83.16	53.53	56.3
Tracktor++ [3]	65.43	**52.71**	**83.09**	58.29	**59.5**
CorrTrack + ReID [10]	64.19	51.33	82.80	56.95	52.0
Ours	**66.89**	51.81	82.71	**58.42**	55.3

Table 2. Comparison to MOT methods on the PoseTrack21-MOT dataset.

is done for two reasons. Firstly, the final assignment might be wrong and, secondly, we do not want to update the track embedding when the detection embedding is noisy due to occlusion or motion blur. To prevent track embeddings from being updated by noisy or low-confident detections, we employ a confidence-guided update of the track embeddings. Let $\mathcal{A} = \{A^n_{:,:-1}\}_{n=1}^N$ be the set of dual-source attention weights of all N decoder stages without the last column. As the number of detections per frame is dynamic, we apply max pooling on the rows of each attention matrix to obtain the maximum attention score for each track and estimate an importance weight w_j with a linear layer as

$$\widehat{\mathcal{A}}_j = \text{concat}\{\max_i A^n_{ji} | n \in [1, N]\} \quad (6)$$

$$w_j = \sigma\left(\sum_n w_n \cdot \widehat{\mathcal{A}}^n_j + b_n\right), \quad (7)$$

where σ is the sigmoid function and $\widehat{A}^n_j$ the maximum attention for track j at layer n. Ultimately, we update the embedding for each track j as a confidence-guided moving average by using the importance weight w_j following $E^t_{T_j} = (1 - w_j) \cdot E^{t-1}_{T_j} + w_j \cdot \widehat{E}^t_{T_j}$. As we will show in our experiments, the confidence-guided embedding update improves the performance compared to solely relying on the *Track Embedding Head* (Fig. 2c) while only adding a negligible overhead.

3.3. Embeddings E^t_D and $E_{T\to D}$

We finally describe how the detection embeddings E^t_D (Fig. 2a) and pose similarity embeddings $E_{T\to D}$ (Fig. 2b) are computed.

Detection Embeddings E^t_D: The embeddings that we extract from each newly detected person need to be robust to occlusion. Since the keypoint heatmaps $\mathcal{H}^t_D$ of the pose estimator provide an indicator which keypoints are occluded, we condition the embedding E^t_D on $\mathcal{H}^t_D$. Specifically, we use a ResNet50 [16] and propose Spatially Adaptive Pose DEnormalization (SPAPDE) layers, which are inspired by [26], as an extension. Each SPAPDE layer operates on a set of keypoint heatmaps $\mathcal{H}^t_D \in \mathbb{R}^{N\times K\times \frac{256}{s}\times\frac{128}{s}}$ along with the current ResNet features $f \in \mathbb{R}^{N\times C\times \frac{256}{s}\times\frac{128}{s}}$ extracted from the corresponding image crops $\mathcal{I}^t_D \in \mathbb{R}^{Nx3x256x128}$, where N is the number of persons in frame t, s denotes the respective scaling factor, K denotes the number of keypoints and C denotes the number of feature channels.

In contrast to batch normalization [20], the normalized input features are scaled and shifted with respect to the keypoint heatmaps. In particular, SPAPDE computes the modulation parameters β and γ following

$$\gamma = \text{conv}(a), \quad \beta = \text{conv}(a), \quad a = \text{ReLU}(\text{conv}(\mathcal{H}^t_D)), \quad (8)$$

where conv denotes a 3x3-convolution and $\beta, \gamma \in \mathbb{R}^{N\times C\times\frac{256}{s}\times\frac{128}{s}}$. The image features f are then conditioned on the heatmaps $\mathcal{H}^t_D$ as follows. Let $f_{n,c,y,x}$ be the feature value for the detected person n and feature channel c at pixel location (x, y). The SPADPE layer first calculates the mean μ_c and standard deviation σ_c over all persons and pixels of f and then adaptively de-normalizes the image features by

$$\widehat{f}_{n,c,y,x} = \gamma_{n,c,y,x} \cdot \frac{f_{n,c,y,x} - \mu_c}{\sigma_c} + \beta_{n,c,y,x}. \quad (9)$$

We replace every batch normalization layer within ResNet50 by SPAPDE layers and train the network following [25].

Moreover, we use a vanilla N-stage transformer encoder [36] without positional encoding to further disentangle the backbone features of severely occluded persons and noisy pose predictions and generate a set of encoded person features $E^T_D \in \mathcal{R}^{N\times 256}$ as shown in Fig. 2a.

Pose Similarity Embeddings $E_{T\to D}$: The pose similarity features $E^t_{T\to D}$ (Fig. 2b) used in our gated attention architecture are based on similarities between bounding boxes and poses using Intersection over Union (IoU) and Object Keypoint Similarity (OKS), respectively. In this work, we rely on three variants of OKS: the first variant considers keypoints which are present in both poses. While the first variant provides a good measure of keypoint alignment, its expressiveness suffers if two poses only share a small subset of keypoints. For that reason, the remaining two variants consider all keypoints present in one of the two poses, respectively. In order to deal with motion, we use as in [29] a warping function ϕ_W that warps the last observed pose of all tracks into the current frame t. The pose similarity $E_{T\to D}$ between tracks and detections is then computed as

Approach	Online	Val. Set	Detector	MOTA	mAP
STAF [28]	✓	v1	-	60.9	70.4
T CPN++ [43]	✓	v1	Cascade R-CNN [4]	64.0	80.9
MIPAL [19]	✓	v1	-	65.7	74.6
KeyTrack [32]	✓	v1	HTC [7]	66.6	81.6
CorrTrack [29]	✓	v1	Cascade R-CNN [4]	68.8	79.2
TKMRNet [47]	✓	v1	Faster R-CNN FPN DCN [48]	68.9	76.7
CorrTrack [29]	✗	v1	Cascade R-CNN [4]	69.1	79.2
CorrTrack [29]	✓	v2	Cascade R-CNN [4]	63.6	75.9
LTIVA [13]	✓	v2	-	64.7	71.4
CombDet [37]	✗	v2	ResNet-101 SNIPER [31]	68.7	81.5
LDGNN [42]	✓	v2	Faster R-CNN FPN DCN [48]	69.2	77.9
Ours	✓	v2	Cascade R-CNN [4]	64.5	76.4

Table 3. Comparison to the state of the art on PoseTrack 2018 [1]. Two versions of the validation set have been released containing 74 (v1) and 170 (v2) sequences, respectively.

follows:

$$E_{T\to D} = \phi_E\left(\left[IOU(\widehat{\mathcal{B}}_T^{t-1}, \mathcal{B}_D^t) \;||\; OKS(\widehat{\mathcal{P}}_T^{t-1}, \mathcal{P}_D^t)\right]\right). \tag{10}$$

Here, $\widehat{\mathcal{P}}_T^{t-1} = \phi_W(\mathcal{P}_T^{t-1})$ and $\widehat{\mathcal{B}}_T^{t-1} = \phi_W(\mathcal{B}_T^{t-1})$ represent the set of warped track poses and track bounding boxes, respectively. The operator $[\cdot \;||\; \cdot]$ represents concatenation, and ϕ_E denotes the pose similarity embedding head, which consists of three linear layers with LayerNorm [2] and GELU [17].

3.4. Training Objective

In a first step, we train the re-identification network following [25]: We apply the triplet loss [18] and center loss [39] after the last pooling layer of ResNet50 and we employ the cross-entropy loss with label smoothing [35] on the classification layer. Subsequently, we freeze the re-identification network and proceed to train our network.

In our approach, the matching layer is trained using a cross-entropy loss, which is defined as follows:

$$\mathcal{L}_{match} = -\frac{1}{N_D}\sum_i y_i \cdot log(p_{ij}^m) + (1-y_i)\cdot p_{i0}^m, \tag{11}$$

where N_D is the total number of detections, $p_{i,j}^m$ represents the probability of matching the i-th detection to its corresponding ground truth track j, and p_{i0}^m is the probability of not matching the i-th detection to any track. The variable y_i takes the value 1 if the i-th detection is assigned to a ground truth track and 0 otherwise.

Since we discard duplicates after the final matching layer, we generate duplicates during training and allow multiple detection assignments to a single track. Specifically, we use detected poses and ground truth poses in the training process that can share the same person identity. To assign identities to the detected poses during training, we employ OKS-based greedy matching to the ground truth poses. We then utilize a duplicates-aware cross-entropy loss function that operates on the attention weights of the encoder and decoder layers. The loss function is defined as follows:

$\mathcal{L}_{attn}^{dec}$	Encoder	$\mathcal{L}_{attn}^{enc}$	CG-Update	HOTA
				53.04
✓				53.40
✓	✓			53.56
✓	✓	✓		53.61
✓	✓	✓	✓	53.94

Table 4. Impact of several components in our pose tracking network on the tracking performance. CG-Update denotes the Confidence-Guided Track Update as discussed in Section 3.1.

$$\mathcal{L}_{attn} = -\frac{1}{N_T}\sum_j \log(p_j), \quad p_j = \left(\sum_i A_{ji}\mathbb{I}_i(j)\right) + \mathbb{I}_{\nexists i}(j)A_{j0}. \tag{12}$$

N_T represents the total number of tracks and p_j denotes the accumulated matching probability for track j, where A_{ji} is the attention weight of the respective encoder/decoder layer and $\mathbb{I}_i(j)$ is 1 if the identity of the current track j and the detection i are the same, and 0 otherwise. If none of the detections matches, *i.e.*, $\mathbb{I}_{\nexists i}(j)$, we maximize the no-match probability A_{j0}, which is the last column of the attention matrix as discussed in Section 3.1. In other words, we want that A_{ji} is large for the correct assignment if and only if a match exists.

The final objective function is a combination of the cross-entropy loss function for the matching layer ($\mathcal{L}_{match}$) and the duplicate-aware cross-entropy loss functions for each encoder and decoder layer:

$$\mathcal{L} = \mathcal{L}_{match} + \sum_k \mathcal{L}_{attn}^{enc_k} + \sum_k \mathcal{L}_{attn}^{dec_k}, \tag{13}$$

where $\mathcal{L}_{attn}^{enc_k}$ and $\mathcal{L}_{attn}^{dec_k}$ represent the duplicate-aware cross-entropy loss functions for the k-th encoder and decoder layer, respectively.

4. Experiments

4.1. Datasets and Evaluation

We evaluate our work on the PoseTrack datasets [1, 10]. Both datasets are large-scale benchmarks for multi-person pose tracking and contain 593 videos for training and 170 for evaluation. The videos contain various activities and include highly diverse poses and severe occlusions as shown in Fig. 1. Since for PoseTrack 2018 the evaluation server is not anymore available, we only report results on the validation set. Compared to PoseTrack 2018, PoseTrack21 [10] provides more annotations and additional benchmarks for multi-object tracking (MOT) and person search. We thus primarily focus on PoseTrack21 [10] in our experiments. For evaluation, we use *keypoint HOTA* [10]. Keypoint HOTA consists of sub-metrics that measure the *detection accuracy (DetA)*, the *association accuracy (AssA)* and

Re-ID Network	Pose-Conditioned	HOTA
ResNet50 [25]		53.47
SPAPDE (ResNet50)	✓	53.94

Table 5. Impact of the proposed SPAPDE network for re-identification on the overall performance on PoseTrack21.

α-Gate Source	Learnable α	AssA	FragA	DetA	HOTA
Attention Logits O_A & O_E		62.20	60.93	47.20	53.94
Attention Logits O_A & O_E	✓	61.80	60.59	47.15	53.74
Attention Weights S_A & S_E		54.74	50.77	47.14	50.53

Table 6. Impact of using attention weights or attention logits in the α-Gate (5) of the Gated Attention Transformer on the overall performance on PoseTrack21.

the *fragmentation accuracy (FragA)*. In addition, we report results for the *keypoint-based MOTA* metric [1]. Both metrics are evaluated on a keypoint level and then averaged. For completeness, we report the keypoint detection performances in terms of *mean average precision (mAP)*.

We follow common practice [32, 41, 29, 42, 10, 37] and utilize a multi-frame pose estimation approach to compensate for missed detections due to motion blur and occlusions during inference. In particular, we utilize keypoint correspondences as in [29, 10]. In the following, we compare our approach to the state of the art. Implementation details and additional ablation studies are provided as supplementary material.

4.2. Comparison with State of the Art

PoseTrack21: We first evaluate our model with and without gating on the PoseTrack21 validation set and compare the performance to methods proposed in [10] using the keypoint HOTA [10] and the MOTA metrics [1]. The results are shown in Table 1. While the performance *without* gated attention is quite low, our proposed gated attention transformer consistently outperforms existing methods, achieving a HOTA score of 53.94 and a MOTA score of 64.1. Compared to CorrTrack W. ReID, our approach boosts the association accuracy (AssA) and fragmentation accuracy (FragA) by $+1.99\%$ and $+1.27\%$ to 62.20 and 60.93, respectively. Additionally, the detection accuracy (DetA) and the mAP increase to 47.20 and 73.6, respectively. While our approach performs online multi-person pose tracking, it also outperforms the offline approach CorrTrack.

We further evaluate our approach on the PoseTrack21-MOT benchmark and compare the performance to the methods in [10]. As we show in Table 2, our approach consistently outperforms existing methods in terms of AssA (66.89) and HOTA (58.42). Tracktor++ [3] achieves a slightly higher DetA and localization accuracy (LocA), which also results in higher MOTA. While Tracktor++ has been trained on the annotated bounding boxes for MOT, our approach has been trained for pose tracking and we simply generate the bounding boxes from the estimated poses. Consequently, MOT methods achieve a better MOTA score due to higher bounding box detection and localization accuracy. As discussed in [24], HOTA is a better metric than MOTA for MOT.

PoseTrack 2018: The comparison with related works on PoseTrack 2018 is difficult for two reasons: i) The PoseTrack 2018 dataset is not available anymore and it is no longer possible to submit results to the official test server; ii) the validation set was released in two different versions. The first version (v1) contains 74 whereas the second version (v2) contains 170 sequences, respectively. For completeness, we also include results that have been reported for v1 in Table 3, but these numbers are not comparable. The results for CorrTrack [29] show that the version v2 is much more difficult. Compared to CorrTrack, our method improves the MOTA score by 0.9% to 64.5. The pose estimation performance increases from 75.9 to 76.4 in terms of mAP. Our appraoch performs similar to [13] in terms of MOTA while achieving a higher mAP. CombDet [37] achieves a higher accuracy, but it uses a stronger multi-frame person detector and is an offline approach, whereas our approach is an online approach. LDGNN [42] also uses a better multi-frame pose estimator, but the code of the pose estimator is not publicly available.

4.3. Ablation Studies

4.3.1 Evaluation of the Network Architecture

We perform ablation experiments to examine the influence of each building block of our proposed method. All experiments are conducted on the PoseTrack21 dataset.

Loss Terms, Transformer Encoder and Confidence-Guided Track Update: We first evaluate the impact of each component in our tracking model on the keypoint HOTA score [10] in Table 4. We start with the gated attention decoder with matching layer as our base model, which we trained with the matching loss $\mathcal{L}_{match}$ (11). Subsequently, we incrementally activate several components and evaluate their impact on the overall performance. Direct supervision on the gated attention layers using $\mathcal{L}_{attn}^{dec}$ (12) increases the overall performance from 53.04 to 53.40 in terms of HOTA. Using additional transformer encoder layers (Fig. 2a) boosts the performance to 53.56. Using the loss $\mathcal{L}_{attn}^{enc}$ (12) also for the encoder increases HOTA to 53.61. The impact of the Confidence-Guided Track Update (Section 3.2) is shown in the last row. It further increases the tracking performance by 0.33 to a HOTA score of 53.94.

Pose Similarity Embedding Update: To update the pose similarity embeddings in the gated decoder layer, we utilize

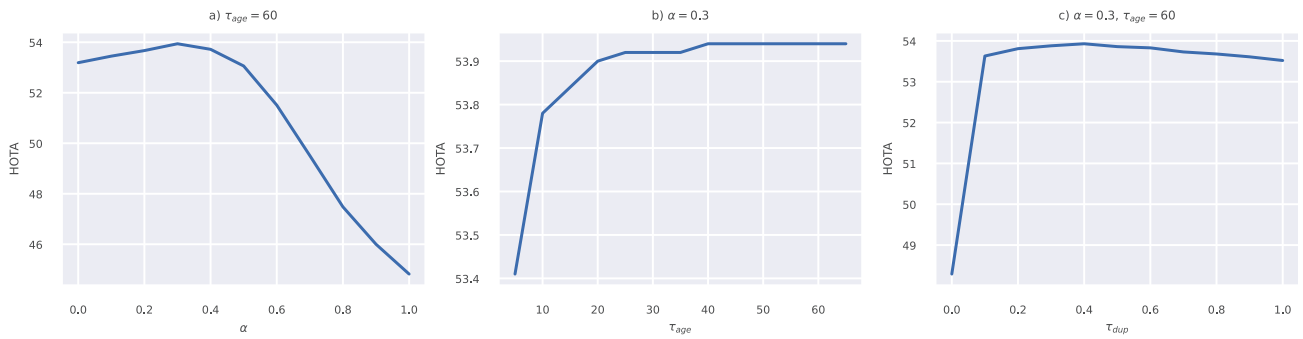

Figure 4. Impact of the parameters α, maximum track ages τ_{age} and the duplicate confidence threshold τ_{dup} on the tracking performance. a) visualizes the impact of α with $\tau_{age} = 60$ and $\tau_{dup} = 0.4$. b) shows the performance evaluation with respect to the maximum track age τ_{age} with $\alpha = 0.3$ and $\tau_{dup} = 0.4$. c) illustrates the impact of τ_{dup} with $\alpha = 0.3$ and $\tau_{age} = 60$.

an α-Gate to weight the contributions of the appearance-based and pose similarity-based attention logits (5). While we evaluate the impact of α in Fig. 4a, Table 6 shows that learning α does not improve the results. We further evaluate the tracking performance when the fusion is not done at the logits, *i.e.*, before the softmax, but after the softmax as in (2), *i.e.*,

$$\widehat{E}^t_{T\to D} = FFN_E\left(A_{:,:-1}\right), \tag{14}$$

where $A_{:,:-1}$ denotes the attention weights without the last column (Section 3.1). The last row in Table 6 shows that the performance largely decreases from a HOTA score of 53.94 to 50.53 in this case.

Re-Identification Model: To evaluate the impact of the re-identification network, we trained our tracking model with two different re-identification networks, which we previously trained on PoseTrack21. We used the re-identification network from [25] and the proposed pose-conditioned SPAPDE network (Section 3.3). Table 5 shows that adding Spatially Adaptive Pose DE-normalization layers (SPAPDE) to the network increases HOTA from 53.47 to 53.94.

4.3.2 Hyperparameter Evaluation

If not otherwise specified, we use in all experiments $\alpha = 0.3$, $\tau_{age} = 60$ and $\tau_{dup} = 0.4$. We finally evaluate the impact of these parameters on the PoseTrack21 dataset.

Impact of α: We evaluate the impact of α in the α-Gate, (3) and (5), in Fig. 4a. α weights the contribution of the appearance-based attention weights and the pose-based attention weights, where $\alpha = 1.0$ only considers appearance-based attention weights and, vice versa, $\alpha = 0.0$ only considers pose similarity-based attention weights. As we can observe in Fig. 4a, the tracking performance drastically decreases for $\alpha > 0.4$. Appearance-based person features are very sensitive to persons with similar appearance as it is common in sports videos, resulting in false associations and a high degree of identity switches. For $0 \leq \alpha \leq 0.3$, we can observe a linear increase in the overall performance, peaking at $\alpha = 0.3$. This shows that pose-based and appearance-based similarities complement each other. Pose similarities provide strong guidance between consecutive frames, while appearance-based features allow to recover inactive tracks, *e.g.*, due to occlusion.

Impact of τ_{age}: We close tracks that have not been tracked for more than τ_{age} frames and do not include them for the detection-to-track matching anymore. Fig. 4b shows that the accuracy saturates at $\tau_{age} = 40$.

Impact of τ_{dup}: During tracking, we remove unmatched detections if they have a matching confidence $m_{ij} > \tau_{dup}$ with an already matched track (Section 3.1). Fig. 4c shows that the accuracy drops without such a threshold since duplicates generate new tracks in this case.

5. Conclusion

We presented a novel gated attention approach for multi-person pose tracking. Our method employs a duplicate-aware association and dynamically adapts via gates the impact of pose-based similarities and appearance-based similarities based on the attention probabilities of each similarity measure. We evaluated our approach on the challenging PoseTrack21 dataset where our approach outperforms previous works for multi-person pose tracking. On PoseTrack 2018, the approach is only outperformed by methods that use a more expensive human pose estimator. We also evaluated the impact of the proposed Spatially Adaptive Pose DEnormalization (SPAPDE) on the pose tracking performance on PoseTrack21, which positively impacts the overall tracking performance.

Acknowledgements This work has been funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - GA 1927/8-1.

References

[1] Mykhaylo Andriluka, Umar Iqbal, Anton Milan, Eldar Insafutdinov, Leonid Pishchulin, Juergen Gall, and Bernt Schiele. PoseTrack: A Benchmark for Human Pose Estimation and Tracking. In *CVPR*, 2018.

[2] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer Normalization, 2016.

[3] Philipp Bergmann, Tim Meinhardt, and Laura Leal-Taixé. Tracking without bells and whistles. In *ICCV*, 2019.

[4] Zhaowei Cai and Nuno Vasconcelos. Cascade R-CNN: High Quality Object Detection and Instance Segmentation. *arXiv preprint arXiv:1906.09756*, 2019.

[5] Di Chen, Andreas Doering, Shanshan Zhang, Jian Yang, Juergen Gall, and Bernt Schiele. Keypoint message passing for video-based person re-identification. In *AAAI*, 2021.

[6] Dapeng Chen, Hongsheng Li, Tong Xiao, Shuai Yi, and Xiaogang Wang. Video person re-identification with competitive snippet-similarity aggregation and co-attentive snippet embedding. *CVPR*, 2018.

[7] Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. Hybrid Task Cascade for Instance Segmentation. *CVPR*, 2019.

[8] Weihua Chen, Xianzhe Xu, Jian Jia, Hao Luo, Yaohua Wang, Fan Wang, Rong Jin, and Xiuyu Sun. Beyond appearance: a semantic controllable self-supervised learning framework for human-centric visual tasks. In *CVPR*, 2023.

[9] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1959.

[10] Andreas Doering, Di Chen, Shanshan Zhang, Bernt Schiele, and Juergen Gall. PoseTrack21: A Dataset for Person Search, Multi-Object Tracking and Multi-Person Pose Tracking. In *CVPR*, 2022.

[11] Andreas Doering, Umar Iqbal, and Juergen Gall. Joint Flow: Temporal Flow Fields for Multi Person Tracking. *CVPR*, 2018.

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICCV*, 2021.

[13] Yang Fu, Sifei Liu, Umar Iqbal, Shalini De Mello, Humphrey Shi, and Jan Kautz. Learning to Track Instances without Video Annotations. *CVPR*, 2021.

[14] Shang Gao, Jingya Wang, Huchuan Lu, and Zimo Liu. Pose-guided Visible Part Matching for Occluded Person ReID. In *CVPR*, 2020.

[15] Rohit Girdhar, Georgia Gkioxari, Lorenzo Torresani, Manohar Paluri, and Du Tran. Detect-and-Track: Efficient Pose Estimation in Videos. In *CVPR*, 2018.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *CVPR*, 2016.

[17] Dan Hendrycks and Kevin Gimpel. Gaussian Error Linear Units (GELUs). *arXiv-Preprint*, 2016.

[18] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In Defense of the Triplet Loss for Person Re-Identification. *arXiv-Preprint*, 2017.

[19] Jihye Hwang, Jieun Lee, Sungheon Park, and Nojun Kwak. Pose estimator and tracker using temporal flow maps for limbs. In *IJCNN*, 2019.

[20] Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *ICML*, 2015.

[21] Sheng Jin, Wentao Liu, Wanli Ouyang, and Chen Qian. Multi-person Articulated Tracking with Spatial and Temporal Embeddings. *CVPR*, 2019.

[22] Dengjie Li, Siyu Chen, Yujie Zhong, Fan Liang, and Lin Ma. DiP: Learning Discriminative Implicit Parts for Person Re-Identification. In *arXiv-Preprint*, 2022.

[23] Jianing Li, Shiliang Zhang, and Tiejun Huang. Multi-Scale 3D Convolution Network for Video Based Person Re-Identification. In *AAAI*, 2019.

[24] Jonathon Luiten, Aljosa Osep, Patrick Dendorfer, Philip Torr, Andreas Geiger, Laura Leal-Taixé, and Bastian Leibe. HOTA: A Higher Order Metric for Evaluating Multi-Object Tracking. *IJCV*, 2020.

[25] Hao Luo, Youzhi Gu, Xingyu Liao, Shenqi Lai, and Wei Jiang. Bag of Tricks and a Strong Baseline for Deep Person Re-Identification. In *CVPRW*, 2019.

[26] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic Image Synthesis With Spatially-Adaptive Normalization. In *CVPR*, 2019.

[27] Lei Qi, Jing Huo, Lei Wang, Yinghuan Shi, and Yang Gao. MaskReID: A Mask Based Deep Ranking Neural Network for Person Re-identification. *ICME*, 2018.

[28] Yaadhav Raaj, Haroon Idrees, Gines Hidalgo, and Yaser Sheikh. Efficient Online Multi-Person 2D Pose Tracking with Recurrent Spatio-Temporal Affinity Fields. *CVPR*, 2019.

[29] Umer Rafi, Andreas Doering, Bastian Leibe, and Juergen Gall. Self-supervised Keypoint Correspondences for Multi-Person Pose Estimation and Tracking in Videos. In *ECCV*, 2020.

[30] M. Saquib Sarfraz, Arne Schumann, Andreas Eberle, and Rainer Stiefelhagen. A Pose-Sensitive Embedding for Person Re-Identification with Expanded Cross Neighborhood Re-Ranking. *CVPR*, 2018.

[31] Bharat Singh, Mahyar Najibi, and Larry S. Davis. SNIPER: Efficient Multi-Scale Training. *NeurIPS*, 2018.

[32] Michael Snower, Asim Kadav, Farley Lai, and Hans Peter Graf. 15 Keypoints Is All You Need. *CVPR*, 2020.

[33] Yifan Sun, Qin Xu, Yali Li, Chi Zhang, Yikang Li, Shengjin Wang, and Jian Sun. Perceive Where to Focus: Learning Visibility-aware Part-level Features for Partial Person Re-identification. *CVPR*, 2019.

[34] Yifan Sun, Liang Zheng, Yi Yang, Qi Tian, and Shengjin Wang. Beyond Part Models: Person Retrieval with Refined Part Pooling (and A Strong Convolutional Baseline). In *ECCV*, 2018.

[35] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the Inception Architecture for Computer Vision. In *CVPR*, 2016.

[36] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In *NeurIPS*, 2017.

[37] Manchen Wang, Joseph Tighe, and Davide Modolo. Combining detection and tracking for human pose estimation in videos. *CVPR*, 2020.

[38] Zhongdao Wang, Liang Zheng, Yixuan Liu, and Shengjin Wang. Towards Real-Time Multi-Object Tracking. *ECCV*, 2020.

[39] Longhui Wei, Shiliang Zhang, Hantao Yao, Wen Gao, and Qi Tian. GLAD: Global-Local-Alignment Descriptor for Pedestrian Retrieval. *ICM*, 2017.

[40] Mikolaj Wieczorek, Barbara Rychalska, and Jacek Dabrowski. On the unreasonable effectiveness of centroids in image retrieval. In *ICNIP*, 2021.

[41] Bin Xiao, Haiping Wu, and Yichen Wei. Simple Baselines for Human Pose Estimation and Tracking. *ECCV*, 2018.

[42] Yiding Yang, Zhou Ren, Haoxiang Li, Chunluan Zhou, Xinchao Wang, and Gang Hua. Learning Dynamics via Graph Neural Networks for Human Pose Estimation and Tracking. *CVPR*, 2021.

[43] Dongdong Yu, Kai Su, Jia Sun, and Changhu Wang. Multi-person Pose Estimation for Pose Tracking with Enhanced Cascaded Pyramid Network. In *ECCVW*, 2018.

[44] Rui Zhang, Zheng Zhu, Peng Li, Rui Wu, Chaoxu Guo, Guan Huang, and Hailun Xia. Exploiting Offset-guided Network for Pose Estimation and Tracking. In *CVPRW*, 2019.

[45] Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenjun Zeng, and Wenyu Liu. FairMOT: On the Fairness of Detection and Re-Identification in Multiple Object Tracking. *ICCV*, 2021.

[46] Liang Zheng, Yujia Huang, Huchuan Lu, and Yi Yang. Pose-Invariant Embedding for Deep Person Re-Identification. *IEEE Transactions on Image Processing*, 2019.

[47] Chunluan Zhou, Zhou Ren, and Gang Hua. Temporal keypoint matching and refinement network for pose estimation and tracking. In *ECCV*, 2020.

[48] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable ConvNets v2: More Deformable, Better Results. *arXiv preprint arXiv:1811.11168*, 2018.