Multi-person Articulated Tracking with Spatial and Temporal Embeddings

Sheng Jin[1,2]   Wentao Liu[1]   Wanli Ouyang[3,4]   Chen Qian [1]

[1] SenseTime Research   [2] Tsinghua University   [3] The University of Sydney

[4] SenseTime Computer Vision Research Group, Australia

[1]{jinsheng, qianchen}@sensetime.com, liuwtwinter@gmail.com   [3] wanli.ouyang@sydney.edu.au

Abstract

We propose a unified framework for multi-person pose estimation and tracking. Our framework consists of two main components, i.e. SpatialNet and TemporalNet. The SpatialNet accomplishes body part detection and part-level data association in a single frame, while the TemporalNet groups human instances in consecutive frames into trajectories. Specifically, besides body part detection heatmaps, SpatialNet also predicts the Keypoint Embedding (KE) and Spatial Instance Embedding (SIE) for body part association. We model the grouping procedure into a differentiable Pose-Guided Grouping (PGG) module to make the whole part detection and grouping pipeline fully end-to-end trainable. TemporalNet extends the spatial grouping of keypoints to temporal grouping of human instances. Given human proposals from two consecutive frames, TemporalNet exploits both appearance features encoded in Human Embedding (HE) and temporally consistent geometric features embodied in Temporal Instance Embedding (TIE) for robust tracking. Extensive experiments demonstrate the effectiveness of our proposed model. Remarkably, we demonstrate substantial improvements over the state-of-the-art pose tracking method from 65.4% to 71.8% Multi-Object Tracking Accuracy (MOTA) on the ICCV'17 Pose-Track Dataset.

1. Introduction

Multi-person articulated tracking aims at predicting the body parts of each person and associating them across temporal periods. It has stimulated much research interest because of its importance in various applications such as video understanding and action recognition [5]. In recent years, significant progress has been made in single frame human pose estimation [3, 9, 12, 25]. However, multi-person articulated tracking in complex videos remains challenging. Videos may contain a varying number of interacting people with frequent body part occlusion, fast body motion, large pose changes, and scale variation. Camera movement and
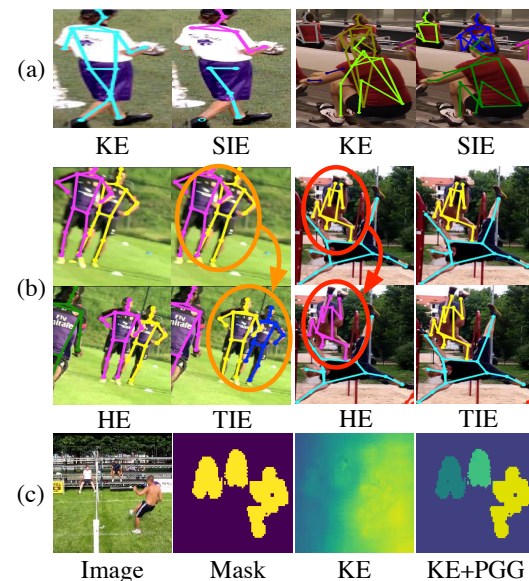

Figure 1. (a) Pose estimation with KE or SIE. SIE may over-segment a single pose into several parts (column 2), while KE may erroneously group far-away body parts together (column 3). (b) Pose tracking with HE or TIE. Poses are color-coded by predicted track ids and errors are highlighted by eclipses. TIE is not robust to camera zooming and movement (column 2), while HE is not robust to human pose changes (column 3). (c) Effect of PGG module. Comparing KE before/after PGG (column 3/4), PGG makes embeddings more compact and accurate, where pixels with similar color have a higher confidence of belonging to the same person.

zooming further pose challenges to this problem.

Pose tracking [14] can be viewed as a hierarchical detection and grouping problem. At the part level, body parts are detected and grouped spatially into human instances in every single frame. At the human level, the detected human instances are grouped temporally into trajectories.

Embedding can be viewed as a kind of permutation-invariant instance label to distinguish different instances. Body parts with similar embeddings have the higher possibility of belonging to the same person. Previous works [21] perform keypoint grouping with Keypoint Em-

bedding (KE). KE [21] is a set of 1-D appearance embedding maps where joints of the same person have similar embedding values and those of different people have dissimilar ones. However, due to the over-flexibility of the embedding space, such representations are difficult to interpret and hard to learn [24]. Arguably, a more natural way for the human to assign ids to targets is by counting in a specific order (from left to right and/or from top to bottom). This inspires us to enforce geometric ordering constraints on the embedding space to facilitate training. Specifically, we add six auxiliary ordinal-relation prediction tasks for faster convergence and better interpretation of KE by encoding the knowledge of geometric ordering. Recently, Spatial Instance Embedding (SIE) [23, 24] is introduced for keypoint grouping. SIE is a 2-D embedding map, where each pixel is encoded with the predicted human center location (x, y). Fig. 1(a) illustrates the typical error patterns of pose estimation with KE or SIE. SIE may over-segment a single pose into several parts (column 2), while KE sometimes erroneously groups far-away body parts together (column 3). KE better preserves intra-class consistency but has difficulty in separating instances for lack of geometric constraints. Since KE captures appearance features while SIE extracts geometric information, they are naturally complementary to each other. Thus, we combine them to achieve better grouping results.

In this paper, we propose to extend the idea of using appearance and geometric information in a single frame to the temporal grouping of human instances for pose tracking. Previous pose tracking algorithms mostly rely on task-agnostic similarity metrics such as the Object Keypoint Similarity (OKS) [34, 36] and Intersection over Union (IoU) [8]. However, such simple geometric cues are not robust to fast body motion, pose changes, camera movement and zoom. For robust pose tracking, we extend the idea of part-level spatial grouping to human-level temporal grouping. Specifically, we extend KE to Human Embedding (HE) for capturing holistic appearance features and extend SIE to Temporal Instance Embedding (TIE) for achieving temporal consistency. Intuitively, appearance features encoded by HE are more robust to fast motion, camera movement and zoom, while temporal information embodied in TIE is more robust to body pose changes and occlusion. We propose a novel TemporalNet to enjoy the best of both worlds. Fig. 1(b) demonstrates typical error patterns of pose tracking with HE or TIE. HE exploits scale-invariant appearance features which are robust to camera zooming and movement (column 1), and TIE preserves temporal consistency which is robust to human pose changes (column 4).

Bottom-up pose estimation methods follow the two-stage pipeline to generate body part proposals at the first stage and group them into individuals in the second stage. Since the grouping is mainly used as post-processing, *i.e.* graph based optimization [11, 12, 14, 16, 27] or heuris-

tic parsing [3, 24], no error signals from the grouping results are back-propagated. We instead propose a fully differentiable Pose-Guided Grouping (PGG) module, making detection-grouping fully end-to-end trainable. We are able to directly supervise the grouping results and the grouping loss is back-propagated to the low-level feature learning stages. This enables more effective feature learning by paying more attention to the mistakenly grouped body parts. Moreover, to obtain accurate regression results, post-processing clustering [23] or extra refinement [24] are required. Our PGG helps to produce accurate embeddings (see Fig. 1(c)). To improve the pose tracking accuracy, we further extend PGG to the temporal grouping of TIE.

In this work, we aim at unifying pose estimation and tracking in a single framework. SpatialNet detects body parts in a single frame and performs part-level spatial grouping to obtain body poses. TemporalNet accomplishes human-level temporal grouping in consecutive frames to track targets across time. These two modules share the feature extraction layers to make a more efficient inference.

The main contributions are summarized as follows:

- For pose tracking, we extend the KE and SIE in still images to Human Embedding (HE) and Temporal Instance Embeddings (TIE) in videos. HE captures human-level global appearance features to avoid drifting in camera motion, while TIE provides smoother geometric features to obtain temporal consistency.

- A fully differentiable Pose-Guided Grouping (PGG) module for both pose estimation and tracking, which enables the detection and grouping to be fully end-to-end trainable. The introduction of PGG and its grouping loss significantly improves the spatial/temporal embedding prediction accuracy.

2. Related Work

2.1. Multi-person Pose Estimation in Images

Recent multi-person pose estimation approaches can be classified into top-down and bottom-up methods. **Top-down** methods [7, 9, 34, 25] locate each person with a bounding box then apply single-person pose estimation. They mainly differ in the choices of human detectors [29] and single-person pose estimators [22, 33, 20]. They highly rely on the object detector and may fail in cluttered scenes, occlusion, person-to-person interaction, or rare poses. More importantly, top-down methods perform single-person pose estimation individually for each human candidate. Thus, its inference time is proportional to the number of people, making it hard for achieving real-time performance. Additionally, the interface between human detection and pose estimation is non-differentiable, making it difficult to train in an end-to-end manner. **Bottom-up** approaches [3, 12, 27]
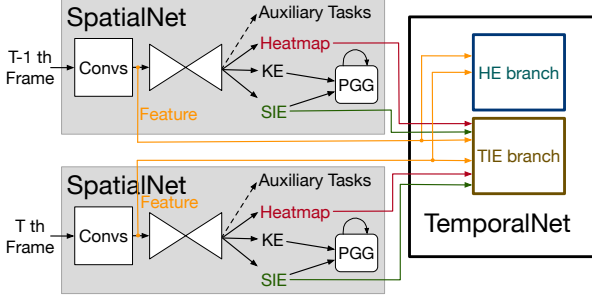
Figure 2. The overview of our framework for pose tracking.

detect body part candidates and group them into individuals. Graph-cut based methods [12, 27] formulate grouping as solving a graph partitioning based optimization problem, while [3, 24] utilize the heuristic greedy parsing algorithm to speed up decoding. However, these bottom-up approaches only use grouping as post-processing and no error signals from grouping results are back-propagated.

More recently, efforts have been devoted to end-to-end training or joint optimization. For top-down methods, Xie *et al.* [35] proposes a reinforcement learning agent to bridge the object detector and the pose estimator. For bottom-up methods, Newell *et al.* [21] proposes the keypoint embedding (KE) to tag instances and train by pairwise losses. Our framework is a bottom-up method inspired by [21]. [21] supervises the grouping in an indirect way. It trains keypoint embedding descriptors to ease the post-processing grouping. However, no direct supervision on grouping results is provided. Even if the pairwise loss of KE is low, it is still possible to produce wrong grouping results, but [21] does not model such grouping loss. We instead propose a differentiable Pose-Guided Grouping (PGG) module to learn to group body parts, making the whole pipeline fully end-to-end trainable, yielding significant improvement in pose estimation and tracking.

Our work is also related to [23, 24], where spatial instance embeddings (SIE) are introduced to aid body part grouping. However, due to lack of grouping supervision, their embeddings are always noisy [23, 24] and additional clustering [23] or refinement [24] is required. We instead employ PGG and additional grouping losses to learn to group SIE, making it end-to-end trainable while resulting in much more compact embedding representation.

2.2. Multi-person Pose Tracking

Recent works on multi-person pose tracking mostly follow the tracking-by-detection paradigm, in which human body parts are first detected in each frame, then data association is performed over time to form trajectories.

Offline pose tracking methods take future frames into consideration, allowing for more robust predictions but having high computational complexity. ProTracker [8] employs 3D Mask R-CNN to improve the estimation of body parts by leveraging temporal context encoded within a sliding temporal window. Graph partitioning based methods [11, 14, 16] formulate multi-person pose tracking into an integer linear programming (ILP) problem and solve spatial-temporal grouping. Such methods achieve competitive performance in complex videos by enforcing long-range temporal consistency.

Our approach is an online pose tracking approach, which is faster and fits for practical applications. Online pose tracking methods [6, 26, 38, 34] mainly use bi-partite graph matching to assign targets in the current frame to existing trajectories. However, they only consider part-level geometric information and ignore global appearance features. When faced with fast pose motion and camera movement, such geometrical trackers are prone to tracking errors. We propose to extend SpatialNet to TemporalNet to capture both appearance features in HE and temporal coherence in TIE, resulting in much better tracking performance.

3. Method

As demonstrated in Figure 2, we unify pose estimation and tracking in a single framework. Our framework consists of two major components: SpatialNet and TemporalNet.

SpatialNet tackles multi-person pose estimation by body part detection and part-level spatial grouping. It processes a single frame at a time. Given a frame, SpatialNet produces heatmaps, keypoint embedding (KE), spatial instance embedding (SIE) and geometric-ordinal maps simultaneously. Heatmaps model the body part locations. KE encodes the part-level appearance features, while SIE captures the geometric information about human centers. The auxiliary geometric-ordinal maps enforce ordering constraints on the embedding space to facilitate training of KE. PGG is utilized to make both KE and SIE be more compact and discriminative. We finally generate the body pose proposals by greedy decoding following [21].

TemporalNet extends SpatialNet to deal with online human-level temporal grouping. It consists of HE branch and TIE branch, and shares the same low-level feature extraction layers with SpatialNet. Given body pose proposals, HE branch extracts region-specific embedding (HE) for each human instance. TIE branch exploits the temporally coherent geometric embedding (TIE). Given HE and TIE as pairwise potentials, a simple bipartite graph matching problem is solved to generate pose trajectories.

3.1. SpatialNet: Part-level Spatial Grouping

Throughout the paper, we use following notations. Let $p = (x, y) \in \mathbb{R}^2$ be the 2-D position in an image, and $p_{j,k} \in \mathbb{R}^2$ the location of body part j for person k. We use $P_k = \{p_{j,k}\}_{j=1:J}$ to represent the body pose of the kth person. We use 2D Gaussian confidence heatmaps to model the body part locations. Let $C_{j,k}$ be the confidence

heatmap for the jth body part of kth person, which is calculated by $C_{j,k}(p) = \exp(-\|p - p_{j,k}\|_2^2/\sigma^2)$ for each position p in the image, where σ is set as 2 in the experiments. Following [3], we take the maximum of the confidence heatmaps to get the ground truth confidence heatmap, i.e. $C_j^*(p) = \max_k C_{j,k}^*(p)$.

The detection loss is calculated by weighted ℓ_2 distance respect to the ground truth confidence heatmaps.

$$L_{det} = \sum_j \sum_p \|C_j^*(p) - C_j(p)\|_2^2. \quad (1)$$

3.1.1 Keypoint Embedding (KE) with auxiliary tasks

We follow [21] to produce the keypoint embedding $\mathcal{K}$ for each type of body part. However, such kind of embedding representation has several drawbacks. First, the embedding is difficult to interpret [21, 24]. Second, it is hard to learn due to its over-flexibility with no direct supervision available. To overcome these drawbacks, we introduce several auxiliary tasks to facilitate training and improve interpretation. The idea of auxiliary learning [32] has shown effective both in supervised learning [28] and reinforcement learning [15]. Here, we explore auxiliary training in the context of keypoint embedding representation learning.

By auxiliary training, we explicitly enforce the embedding maps to learn geometric ordinal relations. Specifically, we define six auxiliary tasks: to predict the 'left-to-right' *l2r*, 'right-to-left' *r2l*, 'top-to-bottom' *t2b*, 'bottom-to-top' *b2t*, 'far-to-near' *f2n* and 'near-to-far' *n2f* orders of human instances in a single image. For example, in the 'left-to-right' map, the person from left to right in the images should have low to high order (value). Fig. 4 (c)(d)(e) visualize some example predictions of the auxiliary tasks. We see human instances are clearly arranged in the corresponding geometric ordering. We also observe that KE (Fig. 4 (b)) and the geometric ordinal-relation maps (c)(d)(e) share some similar patterns, which suggests that KE acquires some knowledge of geometric ordering.

Following [21], $\mathcal{K}$ is trained with pairwise grouping loss $L_{KE} = L_{pull} + L_{push}$. The pull loss (Eq. 2) is computed as the squared distance between the human reference embedding and the predicted embedding of each joint. The push loss (Eq. 3) is calculated between different reference embeddings, which exponentially drops to zero as the increase of embedding difference. Formally, we define the reference embedding for the kth person as $\bar{m}_{\cdot,k} = \frac{1}{J} \sum_j m_j(p_{j,k})$.

$$L_{pull} = \frac{1}{J \cdot K} \sum_k \sum_j \|m(p_{j,k}) - \bar{m}_{\cdot,k}\|_2^2. \quad (2)$$

$$L_{push} = \frac{1}{K^2} \sum_k \sum_{k'} \exp\{-\frac{1}{2}(\bar{m}_{\cdot,k} - \bar{m}_{\cdot,k'})^2\}. \quad (3)$$

For auxiliary training, we replace the push loss with the ordinal loss but keep the pull loss (Eq. 2) the same.

$$L_{aux} = \frac{1}{K^2} \sum_k \sum_{k'} \log(1 + \exp(Ord * (\bar{m}_{\cdot,k} - \bar{m}_{\cdot,k'})))$$
$$+ \frac{1}{J \cdot K} \sum_k \sum_j \|m(p_{j,k}) - \bar{m}_{\cdot,k}\|_2^2, \quad (4)$$

where $Ord = \{1, -1\}$ indicates the ground-truth order for person k and k'. In *l2r*, *r2l*, *t2b*, and *b2t*, we sort human instances by their centroid locations. For example, in *l2r* , if kth person is on the left of k'th person, then $Ord = 1$, otherwise $Ord = -1$. In *f2n* and *n2f*, we sort them according to the head size $\|p_{headtop,k} - p_{neck,k}\|_2^2$.

3.1.2 Spatial Instance Embedding (SIE)

For lack of geometric information, KE has difficulty in separating instances and tends to erroneously group with distant body parts. To remedy this, we combine KE with SIE to embody instance-wise geometric cues. Concretely, we predict the dense offset spatial vector fields (SVF), where each 2-D vector encodes the relative displacement from the human center to its absolute location p. Fig. 4(f)(g) visualize the spatial vector fields of x-axis and y-axis, which distinguish the left/right sides and upper/lower sides relative to its body center. As shown in Fig. 3, subtracted by its coordinate, SVF can be decoded to SIE in which each pixel is encoded with the human center location.

We denote the spatial vector fields (SVF) by $\hat{S}$, and SIE by S. We use ℓ_1 distance to train SVF, where the ground truth spatial vector is the displacement from the person center to each body part.

$$L_{SIE} = \frac{1}{J \cdot K} \sum_{j=1}^{J} \sum_{k=1}^{K} \|\hat{S}(p_{j,k}) - (p_{j,k} - p_{\cdot,k})\|_1, \quad (5)$$

where $p_{\cdot,k} = \frac{1}{J} \sum_j p_{j,k}$, is the center of person k.

3.2. Pose-Guided Grouping (PGG) Module

In prior bottom-up methods [3, 23, 24], detection and grouping are separated. We reformulate the grouping process into a differentiable Pose-Guided Grouping (PGG) module for end-to-end training. By directly supervising the grouping results, more accurate estimation is obtained.

Our PGG is based on Gaussian Blurring Mean Shift (GBMS) [4] algorithm and inspired by [17], which is originally proposed for segmentation. However, directly applying GBMS in the challenging articulate tracking task is not desirable. First, the complexity of GBMS is $O(n^2)$, where n is the number of feature vectors to group. Direct use of
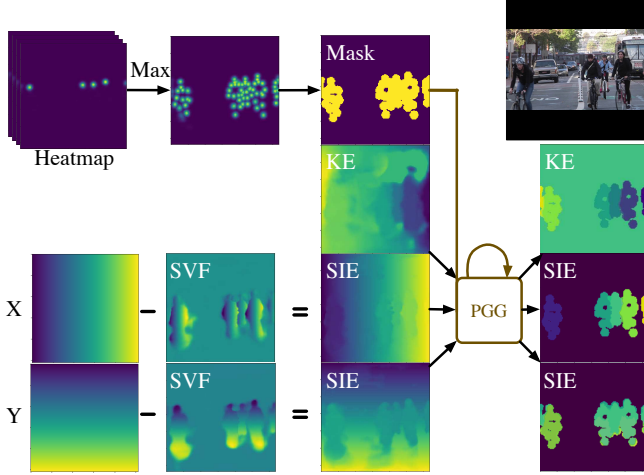
Figure 3. Spatial keypoint grouping with Pose-Guided Grouping (PGG). We obtain more compact and accurate Keypoint Embedding (KE) and Spatial Instance Embedding (SIE) with PGG.

Algorithm 1 Pose-Guided Grouping

Input: KE $\mathcal{K}$, SIE $\mathcal{S}$, Mask $\mathbf{M}$, and iteration number R.
Output: $\mathcal{X}$

1: Concatenate $\mathcal{K}$ and $\mathcal{S}$, mask-selected by $\mathbf{M}$, and reshape to $\mathbf{X}^{(1)} \in \mathbb{R}^{D \times N}$.
2: Initialize $\mathcal{X} = \left[\mathbf{X}^{(1)} \right]$
3: **for** $r = 1, 2, \cdots R$ **do**
4:     Gaussian Affinity $\mathbf{W}^{(r)} \in \mathbb{R}^{N \times N}$. $\mathbf{W}^{(r)}(i,j) = \exp(-\frac{\delta^2}{2} \| x_i^{(r)} - x_j^{(r)} \|_2^2), \forall x_i^{(r)}, x_j^{(r)} \in \mathbf{X}^{(r)}$.
5:     Normalization Matrix. $\mathbf{D}^{(r)} = diag\left(\mathbf{W}^{(r)} \cdot \vec{\mathbf{1}} \right)$
6:     Update. $\mathbf{X}^{(r+1)} = \mathbf{X}^{(r)} \mathbf{W}^{(r)} \left(\mathbf{D}^{(r)} \right)^{-1}$
7:     $\mathcal{X} = \left[\mathcal{X}; \mathbf{X}^{(r+1)} \right]$
8: **end for**
9: **return** $\mathcal{X}$

GBMS on the whole image will lead to huge memory consumption. Second, the predicted embeddings are always noisy especially in background regions, where no supervision is available during training. As illustrated in the top row of Fig. 4, embedding noises exist in the background area (the ceiling or the floor). The noise in these irrelevant regions will affect the mean-shift grouping accuracy. We propose a novel Pose-Guided Grouping module to address the above drawbacks. Considering the sparseness of the matrix (body parts only occupy a small area in images), we propose to use the human pose mask to guide grouping, which rules out irrelevant areas and significantly reduces the memory cost. As shown in Fig. 3, we apply *max* along the channel $\bar{C}(p) = \max_j C_j(p)$ and generate the instance-agnostic pose mask $\mathbf{M} \in \mathbb{R}^{W \times H}$, by thresholding at $\tau = 0.2$. $\mathbf{M}(p)$ is 1 if $\bar{C}(p) > \tau$, otherwise 0.

Both spatial (KE and SIE) and temporal (TIE) embeddings can be grouped by PGG. Take spatial grouping for

example, we refine KE and SIE with PGG module to get more compact and discriminative embedding descriptors. The Pose-Guided Grouping algorithm is summarized in Alg. 1. KE and SIE are first concatenated to $D \times W \times H$ dimensional feature maps. Then embeddings are selected according to the binary pose mask $\mathbf{M}$ and reshaped to $\mathbf{X}^{(1)} \in \mathbb{R}^{D \times N}$ as initialization, where N is the number of non-zero elements in $\mathbf{M}$, ($N \ll W \times H$). Recurrent mean-shift grouping is then applied to $\mathbf{X}^{(1)}$ for R iterations. In each iteration, the Gaussian affinity is first calculated with the isotropic multivariate normal kernel $\mathbf{W} = \exp(-\frac{\delta^2}{2} \| x - x_i \|_2^2)$, where the kernel bandwidth δ is empirically chosen as 5 in the experiments. $\mathbf{W} \in \mathbb{R}^{N \times N}$ can be viewed as the weighted adjacency matrix. The diagonal matrix of affinity row sum $\mathbf{D} = diag(\mathbf{W} \cdot \vec{\mathbf{1}})$ is used for normalization, where $\vec{\mathbf{1}}$ means a vector with all entries one. We then update $\mathbf{X}$ with the normalized Gaussian kernel weighted mean, $\mathbf{X} = \mathbf{X}\mathbf{W}\mathbf{D}^{-1}$. After several iterations of grouping refinement, the embeddings become distinct for heterogeneous pairs and similar for homogeneous ones. When training, we apply the pairwise pull/push losses (Eq. 2 and 3) over all iterations of grouping results $\mathcal{X}$.

3.3. TemporalNet: Human Temporal Grouping

TemporalNet extends SpatialNet to perform human-level temporal grouping in an online manner. Formally, we use the superscript t to distinguish different frames. I^t denotes the input frame at time-step t, which contains K^t persons. SpatialNet is applied to I^t to estimate a set of poses $\mathcal{P}^t = \{ P_1^t, \ldots P_{K^t}^t \}$. TemporalNet aims at temporally grouping human pose proposals $\mathcal{P}^t$ in the current frame with already tracked poses $\mathcal{P}^{t-1}$ in the previous frame. TemporalNet exploits both human-level appearance features (HE) and temporally coherent geometric information (TIE) to calculate the total pose similarity. Finally, we generate the pose trajectories by solving the bipartite graph matching problems, using pose similarity as pairwise potentials.

3.3.1 Human Embedding (HE)

To obtain human-level appearance embedding (HE), we introduce a region-specific HE branch based on [37]. Given predicted pose proposals, HE brach first calculates human bounding boxes to cover the corresponding human keypoints. For each bounding box, ROI-Align pooling [9] is applied to the shared low-level feature maps to extract region-adapted ROI features. The ROI features are then mapped to the human embedding $\mathcal{H} \in \mathbb{R}^{3072}$. HE is trained with triplet loss [31], pulling HE of the same instance closer, and pushing apart embeddings of different instances.

$$L_{HE} = \sum_{\substack{k_1 = k_2 \\ k_1 \neq k_3}} \max(0, \| \mathcal{H}_{k_1} - \mathcal{H}_{k_2} \|_2^2 - \| \mathcal{H}_{k_1} - \mathcal{H}_{k_3} \|_2^2 + \alpha),$$
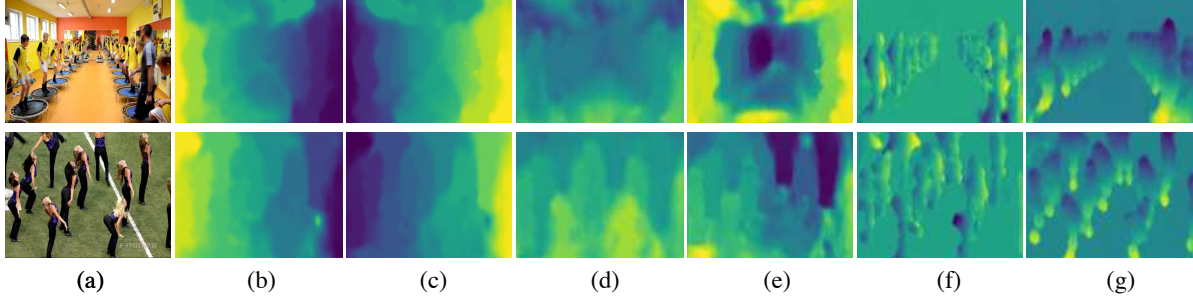
(6)

Figure 4. (a) input image. (b) the average KE. (c)(d)(e) predicted 'left-to-right', 'top-to-bottom' and 'far-to-near' geometric-relation maps. We use colors to indicate the predicted orders, where the brighter color means the higher ordinal value. (f)(g) are the spatial vector fields of x-axis and y-axis respectively. The bright color means positive offset relative to the human center, while dark color means negative.

where the margin term α is set to 0.3 in the experiments.

3.3.2 Temporal Instance Embedding (TIE)

To exploit the temporal information for pose tracking, we naturally extend the Spatial Instance Embedding (SIE) to the Temporal Instance Embedding (TIE). TIE branch concatenates low-level features, body part detection heatmaps and SIE from two neighboring frames. The concatenated feature maps are then mapped to dense TIE.

TIE is a task-specific representation which measures the displacement between the keypoint of one frame and the human center of another frame. This design utilizes the mutual information between keypoint and human in adjacent frames to handle occlusion and pose motion simultaneously. Specifically, we introduce bi-directional temporal vector fields (TVF), which are denoted as $\hat{\mathcal{T}}$ and $\hat{\mathcal{T}}'$ respectively. Forward TVF $\hat{\mathcal{T}}$ encodes the relative displacement from the human center in $(t-1)$-th frame to body parts in the t-th frame, it temporally propagates the human centroid embeddings from $(t-1)$-th to t-th frame. In contrast, Backward TVF $\hat{\mathcal{T}}'$ represents the offset from current t-th frame body center to body parts in the previous frame.

$$L_{TIE} = \frac{1}{J \cdot K^t} \sum_{j=1}^{J} \sum_{k=1}^{K^t} \|\hat{\mathcal{T}}(p_{j,k}^t) - (p_{j,k}^t - p_{.,k}^{t-1})\|_1$$
$$+ \frac{1}{J \cdot K^{t-1}} \sum_{j=1}^{J} \sum_{k'=1}^{K^{t-1}} \|\hat{\mathcal{T}}'(p_{j,k'}^{t-1}) - (p_{j,k'}^{t-1} - p_{.,k'}^t)\|_1,$$
$$(7)$$

where $p_{.,k}^t = \frac{1}{J} \sum_j p_{j,k}^t$, is the center of person k at time step t. Simply subtracted from absolute locations, we get the corresponding Forward TIE $\mathcal{T}$ and Backward TIE $\mathcal{T}'$. Thereby, TIE encodes the temporally propagated human centroid. Likewise, we also extend the idea of spatial grouping to temporal grouping. TemporalNet outputs Forward TIE $\mathcal{T}$ and Backward TIE $\mathcal{T}'$, which are refined by PGG independently. Take Forward TIE $\mathcal{T}$ for example, we generate pose mask M using body heatmaps from the t-th frame. We rule out irrelevant regions of $\mathcal{T}$ and reshape it to $\mathbf{X}^{(1)} \in \mathbb{R}^{D \times N}$. Subsequently, recurrent mean-shift grouping is applied. Again, additional grouping losses (Eq. 2,3) are used to train TIE.

3.3.3 Pose Tracking

The problem of temporal pose association is formulated as a bipartite graph based energy maximization problem. The estimated poses $\mathcal{P}^t$ are then associated with the previous poses $\mathcal{P}^{t-1}$ by bipartite graph matching.

$$\hat{z} = \underset{z}{argmax} \sum_{P_k^t \in \mathcal{P}^t} \sum_{P_{k'}^{t-1} \in \mathcal{P}^{t-1}} \Psi_{P_k^t, P_{k'}^{t-1}} \cdot z_{P_k^t, P_{k'}^{t-1}} \quad (8)$$

$$\text{s.t.} \quad \forall P_k^t \in \mathcal{P}^t, \sum_{P_{k'}^{t-1} \in \mathcal{P}^{t-1}} z_{P_k^t, P_{k'}^{t-1}} \leq 1$$

$$\text{and} \quad \forall P_{k'}^{t-1} \in \mathcal{P}^{t-1}, \sum_{P_k^t \in \mathcal{P}^t} z_{P_k^t, P_{k'}^{t-1}} \leq 1,$$

where $z_{P_k^t, P_{k'}^{t-1}} \in \{0, 1\}$ is a binary variable which implies if the pose hypothesis P_k^t and $P_{k'}^{t-1}$ are associated. The pairwise potentials Ψ represent the similarity between pose hypothesis. $\Psi = \lambda_{HE} \Psi_{HE} + \lambda_{TIE} \Psi_{TIE}$, with Ψ_{HE} for human-level appearance similarity and Ψ_{TIE} for temporal smoothness. λ_{HE} and λ_{TIE} are hyperparameters to balance them, with $\lambda_{HE} = 3$ and $\lambda_{TIE} = 1$.

The human-level appearance similarity is calculated as the ℓ_2 embedding distance: $\Psi_{HE} = \|\mathcal{H}_k - \mathcal{H}_{k'}\|_2^2$. And the temporal smoothness term Ψ_{TIE} is computed as the similarity between the encoded human center locations in SIE $\mathcal{S}$ and the temporally propagated TIE $\mathcal{T}, \mathcal{T}'$.

$$\Psi_{TIE} = \frac{1}{2J} \sum_{j=1}^{J} \left(\|\mathcal{T}'(p_{j,k'}^{t-1}) - \mathcal{S}^t(p_{j,k}^t)\|_2^2 \right.$$
$$\left. + \|\mathcal{T}(p_{j,k}^t) - \mathcal{S}^{t-1}(p_{j,k'}^{t-1})\|_2^2 \right), \quad (9)$$

The bipartite graph matching problem (Eq. 8) is solved using Munkres algorithm to generate pose trajectories.

3.4. Implementation Details

Following [21], SpatialNet uses the 4-stage stacked-hourglass as its backbone. We first train SpatialNet without PGG. The total losses consist of L_{det}, L_{KE}, L_{aux} and L_{SIE}, with their weights 1:1e-3:1e-4:1e-4. We set the initial learning rate to 2e-4 and reduce it to 1e-5 after 250K iterations. Then we fine-tune SpatialNet with PGG included. In practice, we have found the iteration number $R = 1$ is sufficient, and more iterations do not lead to much gain.

TemporalNet uses 1-stage hourglass model [22]. When training, we simply fix SpatialNet and train TemporalNet for another 40 epochs with a learning rate of 2e-4. We randomly select a pair of images I^t and $I^{t'}$ from a range-5 temporal window ($\|t - t'\|_1 \leq 5$) in a video clip as input.

4. Experiments

4.1. Datasets and Evaluation

MS-COCO Dataset [19] contains over 66k images with 150k people and 1.7 million labeled keypoints, for pose estimation in images. For the MS-COCO results, we follow the same train/val split as [21], where a held-out set of 500 training images are used for evaluation.

ICCV'17 PoseTrack Challenge Dataset [13] is a large-scale benchmark for multi-person articulated tracking, which contains 250 video clips for training and 50 sequences of videos for validation.

Evaluation Metrics: We follow [13] to use AP to evaluate multi-person pose estimation and the multi-object tracking accuracy (MOTA) [2] to measure tracking performance.

4.2. Comparisons with the State-of-the-art Methods

We compare our model with the state-of-the-art methods on both pose estimation and tracking on the ICCV'17 Pose-Track validation set. As a common practice [13], additional images from MPII-Pose [1] and MS-COCO [19] are used for training. Table 1 demonstrates our single-frame pose estimation performance. We show that our model achieves the state-of-the-art 77.0 mAP without single-person pose refinement. Table 2 evaluates the multi-person articulated tracking performance. Our model outperforms the state-of-the-art methods by a large margin. Compared with the winner of ICCV'17 PoseTrack Challenge (ProTracker [8]), our method obtains an improvement of 16.6% in MOTA. Our model further improves over the current state-of-the-art pose tracker (FlowTrack [34]) by 6.4% in MOTA with comparable single frame pose estimation accuracy, indicating the effectiveness of our TemporalNet.

4.3. Ablation Study

We extensively evaluate the effect of each component in our framework. Table 3 summarizes the single-frame pose estimation results, and Table 4 the pose tracking results.

Method	Head	Shou	Elb	Wri	Hip	Knee	Ankl	Total
ProTracker [8]	69.6	73.6	60.0	49.1	65.6	58.3	46.0	60.9
PoseFlow [36]	66.7	73.3	68.3	61.1	67.5	67.0	61.3	66.5
BUTDS [16]	79.1	77.3	69.9	58.3	66.2	63.5	54.9	67.8
ArtTrack [13]	78.7	76.2	70.4	62.3	68.1	66.7	58.4	68.7
ML_Lab [38]	**83.8**	**84.9**	76.2	64.0	72.2	64.5	56.6	72.6
FlowTrack [34]	81.7	83.4	**80.0**	**72.4**	75.3	**74.8**	67.1	76.9
Ours	**83.8**	81.6	77.1	70.0	**77.4**	74.5	**70.8**	77.0

Table 1. Comparisons with the state-of-the-art methods on single-frame pose estimation on ICCV'17 PoseTrack Challenge Dataset.

Method	MOTA Head	MOTA Shou	MOTA Elb	MOTA Wri	MOTA Hip	MOTA Knee	MOTA Ankl	MOTA Total
ArtTrack [13]	66.2	64.2	53.2	43.7	53.0	51.6	41.7	53.4
ProTracker [8]	61.7	65.5	57.3	45.7	54.3	53.1	45.7	55.2
BUTD2 [16]	71.5	70.3	56.3	45.1	55.5	50.8	37.5	56.4
PoseFlow [36]	59.8	67.0	59.8	51.6	60.0	58.4	50.5	58.3
JointFlow [6]	-	-	-	-	-	-	-	59.8
FlowTrack [34]	73.9	75.9	63.7	56.1	65.5	65.1	53.5	65.4
Ours	**78.7**	**79.2**	**71.2**	**61.1**	**74.5**	**69.7**	**64.5**	**71.8**

Table 2. Comparisons with the state-of-the-art methods on multi-person pose tracking on ICCV'17 PoseTrack Challenge Dataset.
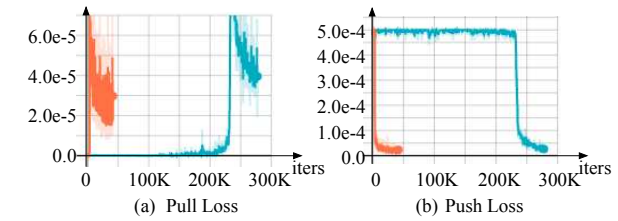
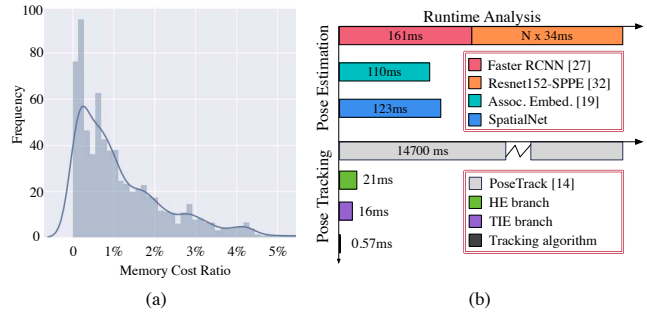Figure 5. Learning curves of keypoint embedding (KE) with (orange) or without (cyan) auxiliary training.

Figure 6. (a) Histogram of the memory cost ratio between PGG and GBMS [4] $\frac{\text{memory cost of PGG}}{\text{memory cost of GBMS}}$ on the PoseTrack val set. Using the instance-agnostic pose mask, PGG reduces the memory consumption to about 1%, *i.e.* 100 times more efficient. (b) Runtime analysis. CNN processing time is measured on one GTX-1060 GPU, while PoseTrack [14] and our tracking algorithm is tested on a single core of a 2.4GHz CPU. N denotes the number of people in a frame, which is 5.97 on average for PoseTrack val set.

For pose estimation, we choose [21] as our baseline, which proposes KE for spatial grouping. We also compare with one alternative embedding approach [18] for design justification. In **BBox** [18], instance location information is encoded as the human bounding box (x, y, w, h) at each pixel. The predicted bounding boxes are then used to group keypoints into individuals. However, such representation is hard to learn due to large variations of its embedding space, resulting in worse pose estimation accuracy com-

pared to **KE** and **SIE**. KE provides part-level appearance cues, while SIE encodes the human centroid constraints. When combined together, a large gain is obtained (74.0% vs. 70.9%/71.3%). As shown in Fig. 5, adding auxiliary tasks (**+aux**) dramatically speeds up the training of KE, by enforcing geometric constraints on the embedding space. It also facilitates representation learning and marginally enhances pose estimation. As shown in Table 3, employing **PGG** significantly improves the pose estimation accuracy (2.3% for KE, 3.8% for SIE, and 2.7% for both combined). End-to-end model training and direct grouping supervision together account for the improvement. Additionally, using the instance-agnostic pose mask, the memory consumption is remarkably reduced to about 1%, as shown in Fig. 6(a), demonstrating the efficiency of PGG. Combining both KE and SIE with PGG, further boosts the pose estimation performance to 77.0% mAP.

For pose tracking, we first build a baseline tracker based on KE and/or SIE. It is assumed that KE and SIE change smoothly in consecutive frames, $\mathcal{K}(p_{j,k}^t) \approx \mathcal{K}(p_{j,k}^{t+1})$ and $\mathcal{S}(p_{j,k}^t) \approx \mathcal{S}(p_{j,k}^{t+1})$. Somewhat surprisingly, such a simple tracker already achieves competitive performance, thanks to the rich geometric information contained in KE and SIE. Employing TemporalNet for tracking significantly improves over the baseline tracker, because of the combination of the holistic appearance features of HE and temporal smoothness of TIE. Finally, incorporating spatial-temporal PGG to refine KE, SIE and TIE, further increase the tracking performance (69.2% vs. 71.8% MOTA). We also compare with some widely used alternative tracking metrics, namely Object Keypoint Similarity (**OKS**), Intersection over Union (**IoU**) of persons and DeepMatching (**DM**) [30] for design justification. We find that TemporalNet significantly outperforms other trackers with task-agnostic tracking metrics. OKS only uses keypoints for handling occlusion, while IoU and DM only consider human in handling fast motion. In comparison, we kill two birds with one stone.

MS-COCO Results. Our SpatialNet substantially improves over our baseline [21] on single frame pose estimation on the MS-COCO dataset. For fair comparisons, we use the same train/val split as [21] for evaluation. Table 5 reports both single-scale (*sscale*) and multi-scale (*mscale*) results. Four different scales $\{0.5, 1, 1.5, 2\}$ are used for multi-scale inference. Our *sscale* SpatialNet already achieves competitive performance against *mscale* baseline. By multi-scale inference, we further gain a significant improvement of 3% AP. All reported results are obtained without model ensembling or pose refinement [3, 21].

4.4. Runtime Analysis

Fig. 6 (b) analyzes the runtime performance of pose estimation and tracking. For pose estimation, we compare with

	Head	Shou	Elb	Wri	Hip	Knee	Ankl	Total
BBox [18]	79.3	75.6	67.4	60.2	67.8	61.6	55.8	67.7
KE [21]	79.8	77.7	71.7	63.4	71.4	66.3	61.4	70.9
SIE	81.4	78.8	72.1	64.2	72.2	66.8	61.7	71.3
KE+SIE	82.2	80.1	74.7	67.4	75.1	69.4	64.6	74.0
KE+SIE+*aux*	82.3	80.3	74.9	67.8	75.2	70.1	65.6	74.3
KE+PGG	81.5	80.0	74.0	65.8	73.4	68.3	65.0	73.2
SIE+PGG	83.4	80.6	74.3	67.4	76.0	71.8	67.6	75.1
Ours	**83.8**	**81.6**	**77.1**	**70.0**	**77.4**	**74.5**	**70.8**	**77.0**

Table 3. Ablation study on single-frame pose estimation (AP) on ICCV'17 PoseTrack validation set. *aux* means auxiliary training with geometric ordinal prediction. Ours (KE+SIE+*aux*+PGG) combines KE+SIE+*aux* with PGG for accurate pose estimation.

	MOTA Head	MOTA Shou	MOTA Elb	MOTA Wri	MOTA Hip	MOTA Knee	MOTA Ankl	MOTA Total
OKS	60.1	60.4	54.5	47.1	58.4	57.0	53.7	56.2
IoU	62.5	63.6	54.3	45.5	59.3	53.6	48.6	55.8
DM [30]	62.9	64.0	54.6	45.7	59.6	53.8	48.7	56.1
KE	72.9	73.3	64.6	55.0	68.7	63.0	58.5	65.7
KE+SIE	75.4	76.1	67.0	57.1	70.9	64.4	59.4	67.7
HE	76.0	76.4	67.7	58.1	71.7	65.4	60.5	68.5
TIE	76.2	76.7	67.8	58.4	71.6	65.3	60.4	68.6
HE+TIE	76.9	77.2	68.4	58.6	72.4	66.0	61.2	69.2
Ours	**78.7**	**79.2**	**71.2**	**61.1**	**74.5**	**69.7**	**64.5**	**71.8**

Table 4. Ablation study on multi-person articulated tracking on ICCV'17 PoseTrack validation set. Ours (HE+TIE+PGG) combines HE+TIE with PGG grouping for robust tracking.

	AP	$AP^{.50}$	$AP^{.75}$	AP^M	AP^L
Assoc. Embed. [21] (*sscale*)	0.592	0.816	0.646	0.505	0.725
Assoc. Embed. [21] (*mscale*)	0.654	0.854	0.714	0.601	0.735
Ours (*sscale*)	0.650	0.865	0.714	0.570	**0.781**
Ours (*mscale*)	**0.680**	**0.878**	**0.747**	**0.626**	0.761

Table 5. Multi-human pose estimation performance on the subset of MS-COCO dataset. *mscale* means multi-scale testing.

both top-down and bottom-up [21] approaches. The top-down pose estimator uses Faster RCNN [29] and a ResNet-152 [10] based single person pose estimator (SPPE) [34]. Since it estimates pose for each person independently, the runtime grows proportionally to the number of people.

Compared with [21], our SpatialNet significantly improves the pose estimation accuracy with the increase of limited computational complexity. For pose tracking, we compare with the graph-cut based tracker (PoseTrack [14]) and show the efficiency of TemporalNet.

5. Conclusion

We have presented a unified pose estimation and tracking framework, which is composed of SpatialNet and TemporalNet: SpatialNet tackles body part detection and part-level spatial grouping, while TemporalNet accomplishes the temporal grouping of human instances. We propose to extend KE and SIE in still images to HE appearance features and TIE temporally consistent geometric features in videos for robust online tracking. An effective and efficient Pose-Guided Grouping module is proposed to gain the benefits of full end-to-end learning of pose estimation and tracking.

References

[1] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 7

[2] Keni Bernardin and Rainer Stiefelhagen. Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008. 7

[3] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 2, 3, 4, 8

[4] Miguel A. Carreiraperpinan. Generalised blurring mean-shift algorithms for nonparametric clustering. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008. 4, 7

[5] Guilhem Cheron, Ivan Laptev, and Cordelia Schmid. P-cnn: Pose-based cnn features for action recognition. In *The IEEE International Conference on Computer Vision (ICCV)*, 2015. 1

[6] Andreas Doering, Umar Iqbal, and Juergen Gall. Joint flow: Temporal flow fields for multi person tracking. *arXiv preprint arXiv:1805.04596*, 2018. 3, 7

[7] Haoshu Fang, Shuqin Xie, and Cewu Lu. Rmpe: Regional multi-person pose estimation. *arXiv preprint arXiv:1612.00137*, 2016. 2

[8] Rohit Girdhar, Georgia Gkioxari, Lorenzo Torresani, Manohar Paluri, and Du Tran. Detect-and-track: Efficient pose estimation in videos. *arXiv preprint arXiv:1712.09184*, 2017. 2, 3, 7

[9] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. *arXiv preprint arXiv:1703.06870*, 2017. 1, 2, 5

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 8

[11] Eldar Insafutdinov, Mykhaylo Andriluka, Leonid Pishchulin, Siyu Tang, Evgeny Levinkov, Bjoern Andres, Bernt Schiele, and Saarland Informatics Campus. Arttrack: Articulated multi-person tracking in the wild. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2, 3

[12] Eldar Insafutdinov, Leonid Pishchulin, Bjoern Andres, Mykhaylo Andriluka, and Bernt Schiele. Deepercut: A deeper, stronger, and faster multi-person pose estimation model. In *European Conference on Computer Vision (ECCV)*, 2016. 1, 2, 3

[13] U. Iqbal, A. Milan, M. Andriluka, E. Ensafutdinov, L. Pishchulin, J. Gall, and Schiele B. PoseTrack: A benchmark for human pose estimation and tracking. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 7

[14] Umar Iqbal, Anton Milan, and Juergen Gall. Pose-track: Joint multi-person pose estimation and tracking. *arXiv preprint arXiv:1611.07727*, 2016. 1, 2, 3, 7, 8

[15] Max Jaderberg, Volodymyr Mnih, Wojciech Marian Czarnecki, Tom Schaul, Joel Z Leibo, David Silver, and Koray Kavukcuoglu. Reinforcement learning with unsupervised auxiliary tasks. *arXiv preprint arXiv:1611.05397*, 2016. 4

[16] Sheng Jin, Xujie Ma, Zhipeng Han, Yue Wu, Wei Yang, Wentao Liu, Chen Qian, and Wanli Ouyang. Towards multi-person pose tracking: Bottom-up and top-down methods. In *ICCV PoseTrack Workshop*, 2017. 2, 3, 7

[17] Shu Kong and Charless Fowlkes. Recurrent pixel embedding for instance grouping. *arXiv preprint arXiv:1712.08273*, 2017. 4

[18] Xiaodan Liang, Yunchao Wei, Xiaohui Shen, Jianchao Yang, Liang Lin, and Shuicheng Yan. Proposal-free network for instance-level object segmentation. *arXiv preprint arXiv:1509.02636*, 2015. 7, 8

[19] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision (ECCV)*, 2014. 7

[20] Wentao Liu, Jie Chen, Cheng Li, Chen Qian, Xiao Chu, and Xiaolin Hu. A cascaded inception of inception network with attention modulated feature fusion for human pose estimation. In *The Thirty-Second AAAI Conference on Artificial Intelligence*, 2018. 2

[21] Alejandro Newell, Zhiao Huang, and Jia Deng. Associative embedding: End-to-end learning for joint detection and grouping. In *Advances in Neural Information Processing Systems (NIPS)*, 2017. 1, 2, 3, 4, 7, 8

[22] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European Conference on Computer Vision (ECCV)*, 2016. 2, 7

[23] Xuecheng Nie, Jiashi Feng, Junliang Xing, and Shuicheng Yan. Generative partition networks for multi-person pose estimation. *arXiv preprint arXiv:1705.07422*, 2017. 2, 3, 4

[24] George Papandreou, Tyler Zhu, Liang-Chieh Chen, Spyros Gidaris, Jonathan Tompson, and Kevin Murphy. Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model. *arXiv preprint arXiv:1803.08225*, 2018. 2, 3, 4

[25] George Papandreou, Tyler Zhu, Nori Kanazawa, Alexander Toshev, Jonathan Tompson, Chris Bregler, and Kevin Murphy. Towards accurate multi-person pose estimation in the wild. *arXiv preprint arXiv:1701.01779*, 2017. 1, 2

[26] Christian Payer, Thomas Neff, Horst Bischof, Martin Urschler, and Darko Štern. Simultaneous multi-person detection and single-person pose estimation with a single heatmap regression network. In *ICCV PoseTrack Workshop*, 2017. 3

[27] Leonid Pishchulin, Eldar Insafutdinov, Siyu Tang, Bjoern Andres, Mykhaylo Andriluka, Peter V Gehler, and Bernt Schiele. Deepcut: Joint subset partition and labeling for multi person pose estimation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2, 3

[28] Charles R Qi, Hao Su, Matthias Nießner, Angela Dai, Mengyuan Yan, and Leonidas J Guibas. Volumetric and multi-view cnns for object classification on 3d data. In *The*

IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 4

[29] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2015. 2, 8

[30] Jerome Revaud, Philippe Weinzaepfel, Zaid Harchaoui, and Cordelia Schmid. Deepmatching: Hierarchical deformable dense matching. *International Journal of Computer Vision (IJCV)*, 2015. 8

[31] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 5

[32] Steven C Suddarth and YL Kergosien. Rule-injection hints as a means of improving network performance and learning time. In *Neural Networks*. 1990. 4

[33] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[34] Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking. In *European Conference on Computer Vision (ECCV)*, 2018. 2, 3, 7, 8

[35] Shuqin Xie, Zitian Chen, Chao Xu, and Cewu Lu. Environment upgrade reinforcement learning for non-differentiable multi-stage pipelines. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3

[36] Yuliang Xiu, Jiefeng Li, Haoyu Wang, Yinghong Fang, and Cewu Lu. Pose flow: Efficient online pose tracking. *arXiv preprint arXiv:1802.00977*, 2018. 2, 7

[37] Qian Yu, Xiaobin Chang, Yi-Zhe Song, Tao Xiang, and Timothy M Hospedales. The devil is in the middle: Exploiting mid-level representations for cross-domain instance matching. *arXiv preprint arXiv:1711.08106*, 2017. 5

[38] Xiangyu Zhu, Yingying Jiang, and Zhenbo Luo. Multi-person pose estimation for posetrack with enhanced part affinity fields. In *ICCV PoseTrack Workshop*, 2017. 3, 7