Hierarchical Associative Encoding and Decoding for Bottom-Up Human Pose Estimation

Congju Du, Zengqiang Yan, Han Yu, Li Yu, *Senior Member, IEEE,* and Zixiang Xiong, *Fellow, IEEE*

***Abstract*—Bottom-up human pose estimation decouples computational complexity from the number of people but requires additional operations to match the detected keypoints to each human instance. Existing approaches treat all keypoints equally while ignoring the relationships among keypoints, which in turn limit the performance ceilings. In this work, we propose a hierarchical associative encoding and decoding framework for bottom-up human pose estimation by introducing additional prior knowledge. Specifically, in addition to keypoint-level and instance-level associations, we further divide keypoints into groups and explore group-level associations. This way, prior knowledge is incorporated to determine the keypoint groups for better associative encoding. To deal with complex poses, we introduce a focal pulling loss to focus more on the hard-to-associate keypoints. Moreover, instead of using a pre-defined order for keypoint grouping, we propose a progressive associative decoding method to dynamically determine the order of keypoints for grouping, which helps reduce isolated keypoints. Experimental results on the MS-COCO, CrowdPose and MPII datasets show superior performance of our proposed associative encoding and decoding algorithms. More importantly, we prove, through validation, that hierarchical associative encoding and decoding can be used as a plug-n-play module for performance improvement regardless of backbone architecture. Our source code and pretrained models are available at https://github.com/ducongju/HAE.**

***Index Terms*—Bottom-up human pose estimation, hierarchical associative encoding, progressive associative decoding, associative embedding.**

I. Introduction

HUMAN pose estimation, as a fundamental task to human-centric image and video understanding, has been widely applied to human action recognition [1]–[3], human pose tracking [4]–[6], person re-identification [7]–[9], person image generation [10], [11], and visual fashion analysis [12].

Existing human pose estimation frameworks can be divided into two categories: top-down [13], [14] and bottom-up [15]–[17]. Top-down frameworks first utilize a human detector, such as Faster R-CNN [18], to detect human instances, and then employ a single-person pose estimator to locate keypoints individually. Therefore, top-down frameworks are usually not trained in an end-to-end manner, resulting in computational redundancy due to the extra human detection step. Comparatively, bottom-up frameworks infer the keypoints and their grouping cues (associative encoding) of all the people indiscriminately before assigning the detected keypoints to the corresponding human instances (associative decoding). They are more likely to achieve real-time pose estimation and have become increasingly popular recently.

Associative encoding and decoding, as the most distinctive component of bottom-up frameworks, aims to associate all the detected keypoints with human instances using parsing strategies. Typically, DeepCut [19], DeeperCut [20], and L-JPA [21] formulated the keypoint grouping process as an integer linear program with the Gurobi solver engine. Unfortunately, solving the optimization problem over a fully connected graph, as shown in Fig. 1 (a), is an NP-hard problem, resulting in high computational complexity. As a result, recent bottom-up frameworks cull the edges (*i.e.* keypoint associations) in the fully connected graph to obtain a tree structure, making the optimization function more amenable. Specifically, OpenPose [16] accomplished keypoint grouping with the help of a bipartite graph matching strategy which introduced part affinity fields to encode the position and orientation of limbs as shown in Fig. 1 (b). CenterNet [22] predicted one more center keypoint for each human instance and encoded a center offset vector at each pixel location belonging to the keypoint region. During associative decoding, every refined keypoint was assigned to its closest center keypoint according to the center offset vectors as shown in Fig. 1 (c). Associative embedding (AE) [15] encodes an embedding to each detected keypoint for grouping. In AE, two keypoints are exclusively matched if their pairwise euclidean distance falls within a specific interval as shown in Fig. 1 (d).

(a) DeepCut [19] (b) OpenPose [16] (c) CenterNet [22] (d) AE [15]

Fig. 1. Illustration of keypoint grouping in bottom-up pose estimation.

All above grouping methods completely ignore the prior knowledge of human skeletons, resulting in unreasonable grouping results. The human body can be regarded as a non-rigid structure (composed of 639 muscles, 206 bones, 78 movable joints, and several immovable joints) [23] where spatial correlation among different types of joints varies sig-

This work was supported in part by the National Natural Science Foundation of China under Grant 61871437 and in part by the Natural Science Foundation of Hubei Province of China under Grant 2019CFA022. The computation is completed in the HPC Platform of Huazhong University of Science and Technology. (*Corresponding author: Li Yu.*)

C. Du, Z. Yan, H. Yu and L. Yu are with the School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: ducongju@hust.edu.cn; z_yan@hust.edu.cn; yuhan2019@hust.edu.cn; hustlyu@hust.edu.cn).

Z. Xiong is with the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77483, USA (e-mail: zx@ece.tamu.edu).

Fig. 2. Illustration of association difficulty between different types of keypoints. The positional relationship between the neck and the hips is stable, while the association of the knees and the ankles can be much harder.

nificantly [24]. For example, the distal bone of the ball-and-socket joint is capable of motion around an indefinite number of axes, while the saddle joint cannot. In other words, the associative difficulty between various types of keypoints is different. As shown in Fig. 2, the degree of freedom between the neck and the hips is less than that of the left knee and the left ankle, resulting in less difficulty in grouping the neck and hips. Moreover, the positional relationship between the neck and the left hip is similar to the right hip, providing auxiliary information for association. Therefore, utilizing prior knowledge is helpful for keypoint grouping in human pose estimation.

In this paper, we make full use of the prior knowledge of human skeletons and propose a hierarchical associative model for better human pose estimation. Specifically, a new mapping scheme for associative encoding is proposed to further divide strongly-associated keypoints into the same group and guide the network to assign weights to keypoints at different hierarchies. In terms of associative decoding, different from existing approaches that use a pre-defined associative ordering to loop through all keypoints for grouping, in our work, the keypoint closest to the current associative embedding is dynamically selected for embedding updates and grouping. This way, our proposed progressive associative decoding is much more stable. Experimental results on publicly-available datasets show the superior performance of our proposed associative encoding and decoding methods for human pose estimation. The main contributions our approach are summarized as follows:

- **Hierarchical Associative Encoding (HAE)** for incorporating prior knowledge of human skeletons into human pose estimation, including hierarchical associative model to divide strongly associated keypoints into the same keypoint group, hierarchical associative loss to focus more on the hard-to-associate keypoints between keypoint groups, and focal pulling loss to counter the influence of outlier keypoints.
- **Progressive Associative Decoding (PAD)** with better associative performance and faster pose inference speed. Compared to using a fixed associative order for grouping, PAD is more flexible, especially in dealing with complex poses.
- A plug-n-play module for performance improvement. Through extensive validation, we show that our proposed hierarchical associative encoding and decoding is architecture-agnostic without introducing extra model parameters.

II. Related Work

A. Location encoding and decoding

Standard location encoding [25], [26] transforms ground-truth coordinates to a series of Gaussian response heatmaps for capturing spatial and contextual information. Papandreou *et al.* [27], [28] employed a coarse-to-fine approach to encode binary heatmaps and the corresponding offsets to indicate the regions of keypoints. Du *et al.* [29], [30] presented a scale-sensitive heatmap algorithm to handle the scale variation problem among human instances by modifying the standard deviation, truncated radius, and shape of the Gaussian heatmaps. Zhao *et al.* [14] added a network block to existing pose estimation frameworks, aiming to encode the pose quality. For 3D pose estimation, Wei *et al.* [31] proposed a view-invariant location encoding framework, which automatically transforms the intermediate 3D poses into a consistent view.

Location decoding [26] is usually needed to obtain the final coordinates on the heatmap. Specially, aiming to alleviate the quantization error, Zhang *et al.* [32] proposed a superior distribution-aware coordinate decoding method to predict keypoint locations in the original image coordinate space. Xie *et al.* [33] further refined the decoded coordinates by exploiting temporal coherency among the heatmaps of adjacent frames; their proposed hierarchical dynamic programming module can be inserted into image-based networks.

B. Associative encoding and decoding

Associative encoding can be classified into graph-, limb-, center-, and embedding-based methods. Graph-based methods [19]–[21] encoded all the keypoint pairwise relations and cast them as an optimization problem. HGG [34] developed graph partitioning by using a graph neural network [35] on both edge discriminators and macro-node discriminators to group keypoints. As a typical limb-based method, OpenPose [16] proposed the Part Affinity Fields (PAFs) to associate body parts. For each pixel in a particular limb region, PAFs encoded the location and direction between each keypoint pair. SimplePose [36] simplified PAFs such that body parts only require half the dimensions of PAFs to encode the associative information. PifPaf [37] encoded both scalar and vector fields to form composite fields by predicting one confidence map and two associative vectors for keypoint grouping and two widths for precise regression. PersonLab [28] defined mid-range pairwise offsets to assign keypoints to each human instance. In terms of center-based methods, CenterNet [22] encoded offsets at center keypoints and decoded them by heuristically matching offsets to their closest predicted keypoints. SPM [38] proposed an improved Structured Pose Representation (SPR) by dividing keypoints into hierarchies to unify position information of human instances and body joints. For embedding-based methods, AE [15] predicted a tagmap that encourages pairs of tags to have similar values if the corresponding keypoints belong to the same human instance in ground truth or dissimilar values otherwise. HigherHRNet [39] followed the same AE encoding method but added a deconvolution module based on HRNet.

The above works do not fully leverage the prior knowledge of human skeletons for associative encoding. The idea of

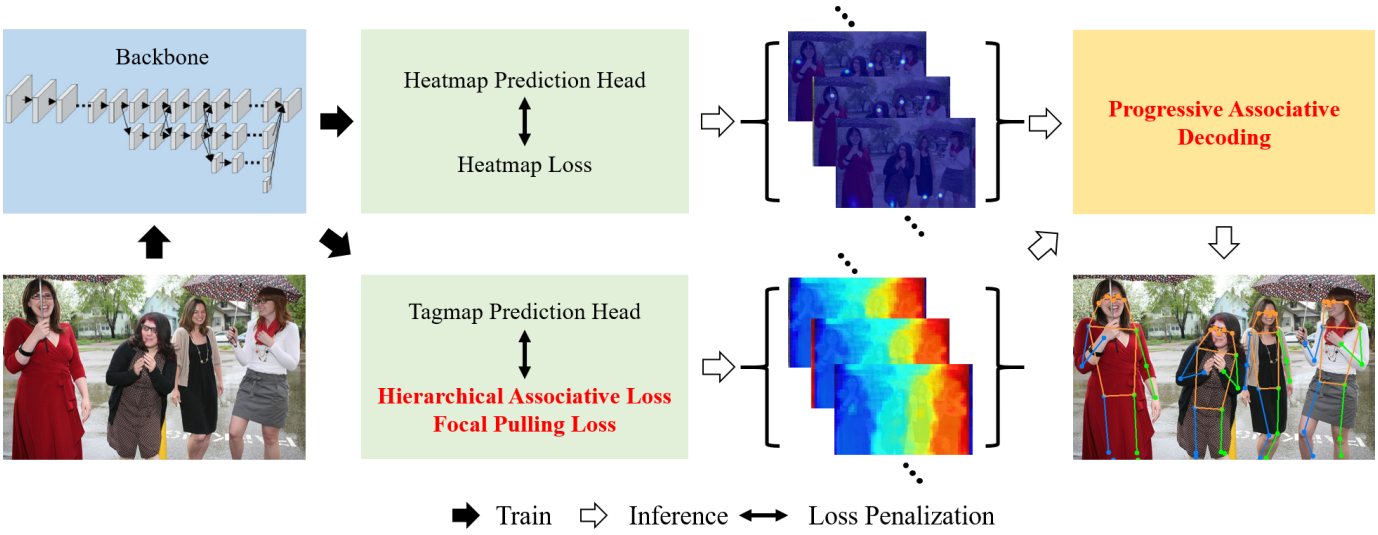

Fig. 3. Overview of the proposed method for multi-person pose estimation. The encoding process consists of standard Gaussian heatmap location encoding and the proposed hierarchical associative encoding, while the decoding process consists of standard location decoding and the proposed progressive associative decoding.

exploiting prior knowledge of human part relationship for parsing human semantics has been extensively studied in the broad field of human-centric perception, such as human parsing [40], [41] and 3D pose estimation [42]. In this paper, we introduce a hierarchical associative model to classify strongly-associated keypoints into the same keypoint group. We adopt embedding-based methods as the associative encoding pipeline.

Associative decoding refers to the strategy of grouping the detected keypoints into human instances. Deepcut [19] solved a sequence of relaxation of the subset partitioning and labeling problem. OpenPose [16] selected a minimal number of edges to obtain a tree skeleton of each human pose rather than using the fully connected graph. In this way, the grouping problem is decomposed into a set of bipartite matching subproblems and addressed by minimal greedy inference with lower computational cost. KHGF [17] utilized the guiding offset fields at the position of a local maximal in the keypoint Gaussian heatmaps to measure the pairing score. Candidate keypoint associations were sorted in descending order on the basis of the scores. PersonLab [28] created a priority queue and popped candidate detections out of the queue in descending order of scores for fast greedy associative decoding. CenterNet [22] associated individual keypoint detection with the closest human instance according to the predicted center offset. AE [15] relied on the order of association to perform maximum matching where the weight of each keypoint was determined according to both the tagmap and heatmap. DSPF [43] formulated joint association as maximum-weight bipartite matching and developed a differentiable solution based on projected gradient descent and Dykstra's cyclic projection algorithm. We propose progressive associative decoding to adjust the order of association adaptively to minimize the influence of inaccurate reference embedding from the previous iteration.

III. Methodology

A. Problem Formulation

Given an input image I, multi-person pose estimation aims to obtain a set of human poses $\mathcal{P}$, where each pose consists of K anatomical joints or landmarks (*i.e.* keypoints). Formally, the n-th human poses can be represented as

$$\mathcal{P}_n = \{p_{n,1}, p_{n,2}, ..., p_{n,K}\}, n \in \{1, 2, ..., N\}, \quad (1)$$

where N is the number of human instances in I. For 2D pose estimation, $p_{n,k} = (x_{n,k}, y_{n,k})$ denotes the coordinates of the k-th keypoint from the n-th human instance. For the 3D case, $p_{n,k}$ is written as $(x_{n,k}, y_{n,k}, z_{n,k})$.

Top-down frameworks first use a human detector f_{det} to crop human instances and then perform single-person pose estimation g_{sgl} to each detected human instance, which can be summarized as

$$\begin{aligned} f_{det} &: I \rightarrow \{i_1, i_2, ..., i_N\}, \\ g_{sgl} &: \{i_1, i_2, ..., i_N\} \rightarrow \mathcal{P}, \end{aligned} \quad (2)$$

where i_n denotes the n-th detected human instance cropped by a set of bounding boxes.

In contrast, bottom-up frameworks first detect all identity-agnostic keypoints by a keypoint estimator g_{mul} and then group them into individual human instances by solving a graph partition problem h_{group}. The encoding and decoding process is as follows

$$\begin{aligned} g_{mul} &: I \rightarrow \{p_1, p_2, ..., p_{N \cdot K}\}, \mathcal{R}, \\ h_{group} &: \{p_1, p_2, ..., p_{N \cdot K}\}, \mathcal{R} \rightarrow \mathcal{P}, \end{aligned} \quad (3)$$

where $\mathcal{R}$ denotes the reference for assigning the identity-agnostic keypoints into human instances. Here, the training processes during $I \rightarrow \{p_1, p_2, ..., p_{N \cdot K}\}$ and $I \rightarrow \mathcal{R}$ are called location encoding and associative encoding, and the inference processes during $\{p_1, p_2, ..., p_{N \cdot K}\} \rightarrow \mathcal{P}$ and $\mathcal{R} \rightarrow \mathcal{P}$ are considered as location decoding and associative decoding, respectively.

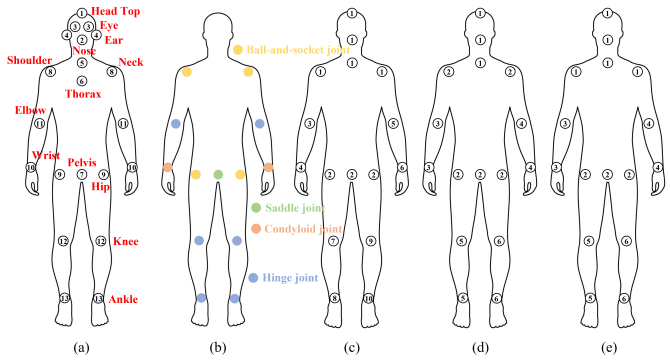

Fig. 4. Illustration of hierarchical association: (a) human keypoints, (b) types of joints in human body, (c) the proposed hierarchical associative model, (d) the hierarchical model based on spatial prior [44], and (e) the hierarchical model based on mutual information [45].

B. Frameworks of Multi-person pose estimation

The overall architecture of our proposed method for human pose estimation is shown in Fig. 3. During training, each input image is first fed to the high-resolution backbone to obtain global features. Then, the location encoding part is trained by the ground-truth Gaussian heatmaps, which can be formulated as

$$\mathcal{H}_k = \sum_{n=1}^{N} \exp\left(-\frac{(x-x_{n,k})^2 + (y-y_{n,k})^2}{2\sigma^2}\right), \quad (4)$$

where (x, y) denotes the 2D pixel coordinates and σ is the standard deviation of the Gaussian kernel.

For associative encoding, the hierarchical associative model first classifies strongly-associated joints into the same keypoint group, forming "keypoint-, group-, and instance-level" hierarchical representations. Then, the hierarchical associative encoding module uses associative embedding to generate appropriate tag values for each keypoint, keypoint group, and human instance. Through a hierarchical associative loss, pairs of keypoints and keypoint groups are encouraged to share similar tag values if they are from the same human instance. To deal with complex poses, a focal pulling loss is introduced to take into account those difficult-to-associate keypoints.

For inference, each image is fed to the proposed hierarchical associative encoding module to produce its heatmap and tagmap separately. Then, standard location decoding is employed to first acquire the maximal activation coordinates $\tilde{p}_{n,k}$ of each k-th keypoint by non-maximum suppression (NMS) and then determine the final coordinates by $\frac{1}{4}$ sub-pixel adjustment toward the direction of its next highest neighbor $\hat{p}_{n,k}$ in the heatmaps

$$p_n^k = \tilde{p}_{n,k} + 0.25\frac{\hat{p}_{n,k} - \tilde{p}_{n,k}}{\|\hat{p}_{n,k} - \tilde{p}_{n,k}\|_2}, \quad (5)$$

where $\|\cdot\|_2$ defines the magnitude of a vector. After that, the corresponding tag values are retrieved in the tagmap, and finally a progressive associative decoding module is adopted for keypoint grouping. More details are given in the sequel.

C. Hierarchical Associative Model

Hierarchical models [44]–[47] aim to capture the global and higher-order relationships among keypoints and describe an exponential number of reasonable poses, allowing pose estimation to satisfy the relational constraints optimally. A manually defined hierarchical model based on the body structure (using fully shared features) [44] is shown in Fig. 4 (d) and a hierarchical model based on mutual information (using specific learned features for related keypoints) [45] is shown in Fig. 4 (e). These models are effective for location encoding in terms of improving localization accuracy but can hardly help associative encoding which focuses more on pose diversity. Based on human anatomy [24], [48], joints with more degrees of freedom are more likely to generate complex and diverse poses and thus more difficult to associate with human instances in multi-person scenarios. According to the types of joints in Fig. 4 (b), there are more complex association situations between hinge joints, hinge joints and condyloid joints, hinge joints and ball-and-socket joints, condyloid joints and ball-and-socket joints. Therefore, we group strongly-associated joints into the same keypoint group according to which hierarchical associative model is constructed as shown in Fig. 4 (c).

Formally, we represent the hierarchical associative model as a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \mathcal{V}_1 \cup \mathcal{V}_2 \cup \mathcal{V}_3$ is the node set in $\mathcal{G}$ which can be represented as three different hierarchies, namely keypoint, group, and instance levels. Take the CrowdPose keypoint annotation as an example where keypoint-level nodes include $\mathcal{V}_1$ = {head-top, neck, left-shoulder, right-shoulder, left-elbow, right-elbow, left-wrist, right-wrist, left-hip, right-hip, left-knee, right-knee, left-ankle, right-ankle}. They can be divided into 10 keypoint groups as $\mathcal{V}_2$={$\mathcal{K}_1, \mathcal{K}_2, ..., \mathcal{K}_{10}$}, where $\mathcal{K}_1$ = {head top, neck, left shoulder, right shoulder}, $\mathcal{K}_2$ = {left hip, right hip}, $\mathcal{K}_3$ = {left wrist}, etc. The edge set $\mathcal{E}$ represents whether a keypoint- or group-level node belongs to a higher-level node. Our hierarchical associative model does not introduce any additional annotation cost because all the higher-level nodes are available from a simple combination of defined keypoints in $\mathcal{V}_1$.

D. Hierarchical Associative Encoding

To correctly assign keypoints to human instances, similar to associative embedding (AE) [15], we first encode an embedding for each detected keypoint as a tag to identify its human instance and then group the keypoints with similar tags into a single human instance during associative decoding. Following AE, the tag values are set as real numbers to encourage keypoints belonging to the same human instance to have similar tag values and otherwise to have dissimilar tag values. Since the number of human instances in each image is unknown, it is of high complexity to generate ground-truth embeddings/tag values. Therefore, according to the Fisher Criterion which minimizes the intra-class scatter and maximizes the inter-class scatter, we propose an unsupervised approach to obtain embeddings. For each individual human instance, a reference embedding is computed by taking the mean of the keypoint embeddings. For keypoints belonging to the same human instance, we minimize the distance between each keypoint's embedding and the reference embedding. Then, the reference

This article has been accepted for publication in IEEE Transactions on Circuits and Systems for Video Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2022.3215564

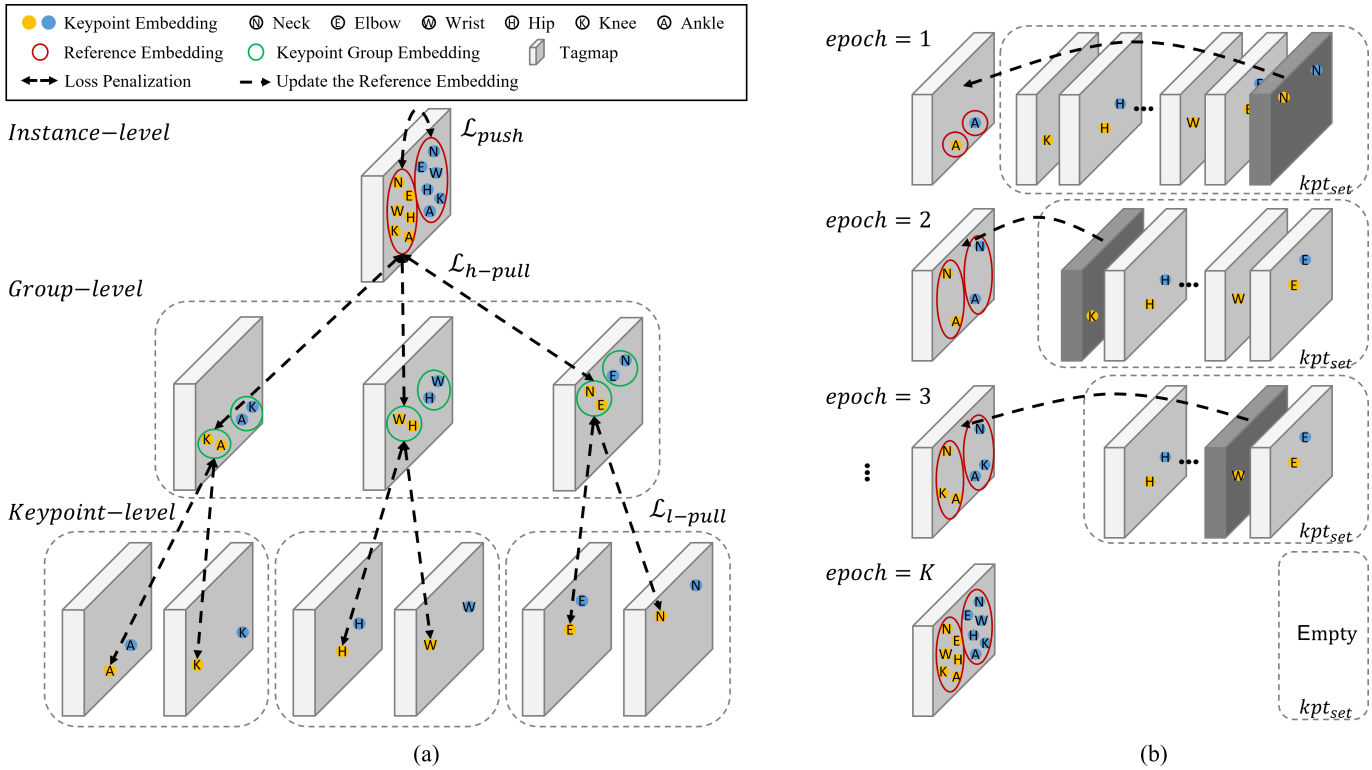

Fig. 5. (a) Example of calculating the hierarchical associative loss. Yellow and blue solid circles refer to keypoint embeddings, red hollow circles refer to reference embeddings, and green hollow circles refer to keypoint group embeddings. We first obtain the corresponding keypoint embeddings from the initialized tag map and implement the partitioning of keypoint-, group-, and instance-level based on the prior knowledge of human skeletons (Fig. 4(c) and Table I). Then the reference embedding and the keypoint group embedding are calculated according to (6) and (8). $\mathcal{L}_{push}$ maximizes the distance between the reference embeddings of different human instances, $\mathcal{L}_{l-pull}$ minimizes the distance between the keypoint embeddings and the keypoint group embeddings, and $\mathcal{L}_{h-pull}$ minimizes the distance between the keypoint group embeddings and the reference embeddings. The overall hierarchical associative loss can be obtained from (11). $\lambda = \frac{K}{G}$ indicates that no prior knowledge is used according to (14). (b) Illustration of the progressive associative decoding algorithm. At each iteration, the keypoint with the smallest distance (absolute value of the difference between the candidate keypoint embeddings and the reference embedding) in the set of candidate keypoints kpt_{set} is selected and added to the calculation of the reference embedding until the set is empty.

embeddings between pairs of individual human instances are maximized.

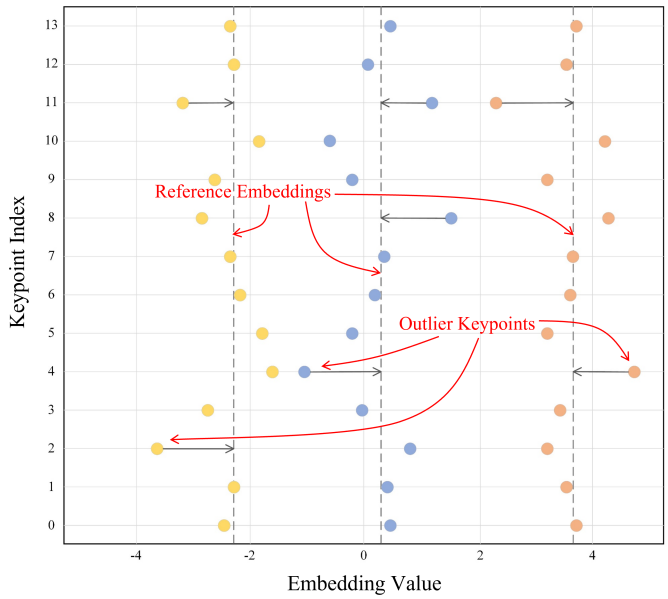

Fig. 6. Exemplar keypoint embeddings where keypoints distant from the reference embeddings are denoted as the outlier keypoints.

Hierarchical associative loss. Fig. 5 (a) illustrates the details of calculating the hierarchical associative loss. Let t_k denote the predicted tagmap for the k-th keypoint, where $t(p)$ is the tag value at pixel location p. Given the ground-truth human poses $\mathcal{P}$ and all K annotated keypoints, the reference embedding for the n-th individual can be calculated as

$$\bar{t}_n = \frac{1}{K} \sum_{k=1}^{K} t_k \left(p_{n,k}\right). \tag{6}$$

To guarantee distinct tag values for different human instances, we maximize the distance between reference embeddings of different human instances. $\mathcal{L}_{push}$ is defined as

$$\mathcal{L}_{push} = \frac{1}{N(N-1)} \sum_{n=1}^{N} \sum_{n' \neq n} \exp\left(-\frac{1}{2\sigma^2} \left(\bar{t}_n - \bar{t}_{n'}\right)^2\right). \tag{7}$$

Considering the case of single-person images ($N = 1$), $\mathcal{L}_{push}$ is set to zero.

As described above, in addition to keypoint- and instance-level associations, our proposed hierarchical associative model further divides keypoints into groups. Given the g-th keypoint

group in the n-th person, its embedding is defined as

$$\bar{t}_{n,g} = \frac{1}{K_g} \sum_{k=1}^{K_g} t_k\left(p_{n,k}\right), \tag{8}$$

where K_g denotes the number of keypoints in the g-th keypoint group that can be acquired according to Fig. 4 (c).

To minimize the distance between keypoint embeddings and keypoint group embeddings, $\mathcal{L}_{l-pull}$ is defined as

$$\mathcal{L}_{l-pull} = \frac{1}{N} \sum_{n=1}^{N} \sum_{g=1}^{G} \sum_{k=1}^{K_g} \left(\bar{t}_{n,g} - t_k\left(p_{n,g,k}\right)\right)^2, \tag{9}$$

where G is the number of keypoint groups in the n-th human instance. Thus, keypoint embeddings belonging to the same keypoint group are penalized to be similar.

Furthermore, to minimize the distance between the keypoint group embeddings belonging to the same human instance, $\mathcal{L}_{h-pull}$ is defined as

$$\mathcal{L}_{h-pull} = \frac{1}{N} \sum_{n=1}^{N} \sum_{g=1}^{G} \left(\bar{t}_n - \bar{t}_{n,g}\right)^2. \tag{10}$$

If there is only one person in the image, $\mathcal{L}_{l-pull}$ and $\mathcal{L}_{h-pull}$ can be simplified as $\sum_{g=1}^{G} \sum_{k=1}^{K_g} \left(\bar{t}_g - t_k\left(p_{g,k}\right)\right)^2$ and $\sum_{g=1}^{G} \left(\bar{t} - \bar{t}_g\right)^2$. The overall hierarchical associative loss is written as

$$\mathcal{L}_{hi} = \mathcal{L}_{push} + \mathcal{L}_{l-pull} + \lambda \mathcal{L}_{h-pull}, \tag{11}$$

where λ is a trade-off parameter that is set as 1.5 in our experiments.

Compared to the classical associative loss defined as

$$\begin{aligned} \mathcal{L}_{ae} = & \frac{1}{N} \sum_{n=1}^{N} \sum_{k=1}^{K} \left(\bar{t}_n - t_k\left(p_{n,k}\right)\right)^2 \\ & + \frac{1}{N(N-1)} \sum_{n=1}^{N} \sum_{n' \neq n} \exp\left(-\frac{1}{2\sigma^2}\left(\bar{t}_n - \bar{t}_{n'}\right)^2\right), \end{aligned} \tag{12}$$

the difference between the hierarchical associative loss and the classical associative loss is

$$\begin{aligned} \mathcal{L}_{hi} - \mathcal{L}_{ae} &= \mathcal{L}_{push} + \mathcal{L}_{l-pull} + \lambda \mathcal{L}_{h-pull} - \mathcal{L}_{ae} \\ &= \frac{1}{N} \sum_{n=1}^{N} \sum_{g=1}^{G} \sum_{k=1}^{K_g} \left(\bar{t}_{n,g} - t_k\left(x_{n,g,k}\right)\right)^2 \\ &\quad + \frac{\lambda}{N} \sum_{n=1}^{N} \sum_{g=1}^{G} \left(\bar{t}_n - \bar{t}_{n,g}\right)^2 \\ &\quad - \frac{1}{N} \sum_{n=1}^{N} \sum_{k=1}^{K} \left(\bar{t}_n - t_k\left(x_{n,k}\right)\right)^2 \\ &= \frac{1}{N} \sum_{n=1}^{N} \left(\sum_{g=1}^{G} \left(\lambda - K_g\right) \bar{t}_{ng}^2 + \left(K - \lambda G\right) \bar{t}_n^2 \right). \end{aligned} \tag{13}$$

When $G = 1$, $\bar{t}_{ng}$ becomes $\bar{t}_n$, resulting in $\mathcal{L}_{hi} - \mathcal{L}_{ae} = 0$. Then, our proposed hierarchical associative loss degenerates to the classical associative loss. When $G = K$, $\bar{t}_{ng}$ becomes $t_k\left(p_{nk}\right)$ and $\mathcal{L}_{hi}$ equals to $\mathcal{L}_{ae}$. Therefore, $\mathcal{L}_{hi}$ is a generalized form of the classical associative loss L_{ae}. In the case when each keypoint group has the same number of keypoints, *i.e.*, $K_g = \frac{K}{G}$. If $1 < G < K$, (13) can be rewritten as

$$\mathcal{L}_{hi} - \mathcal{L}_{ae} = \frac{\lambda - K_g}{N} \sum_{n=1}^{N} \left(\sum_{g=1}^{G} \bar{t}_{ng}^2 - G \bar{t}_n^2 \right). \tag{14}$$

According to the inequality of arithmetic and squared means, we have $\sum_{g=1}^{G} \bar{t}_{ng}^2 - G\bar{t}_n^2 \geq 0$. As the tag values of keypoints are randomly initialized, we can have $\sum_{g=1}^{G} \bar{t}_{ng}^2 - G\bar{t}_n^2 > 0$. Given $\lambda > \frac{K}{G}$, $\mathcal{L}_{hi} - \mathcal{L}_{ae}$ is greater than 0 and thus is more sensitive to the keypoints whose tag values are distant from the keypoint groups. The hierarchical associative model encodes the prior knowledge of the human skeletons that some joint pairs are strongly associated. By adjusting λ, $\mathcal{L}_{hi}$ would focus more on the hard-to-associate keypoints between keypoint groups.

Focal pulling loss. In crowded scenes, there exist highly imbalanced easy and hard samples due to complex poses. Here, hard samples refer to keypoints that are difficult to associate with human instances and tend to obtain outlier tag values during initial training as shown in Fig. 6. Different from the focal L2 loss [17], [36] and online hard keypoints mining (OHKM) [49], which are introduced to up-weight the contribution of the hard-to-detect samples, keypoints distant from the reference embeddings are penalized with higher weights to train the network for complex pose estimation. To detect these hard-to-associate samples (outlier keypoints), we utilize the interquartile range (IQR) as a measure of statistical dispersion and denote the keypoints whose tag values fall beyond the outer fences as hard-to-associate samples. Formally, the embedding array $t_k\left(p_{n,g,k}\right)$ is divided into quartiles denoted as Q_1 (the lower quartile), Q_2 (the median), and Q_3 (the upper quartile), respectively. Then, the IQR is defined as $IQR = Q_3 - Q_1$, and the lower and the upper fences are calculated as $T_{lf} = Q_1 - 1.5 * IQR$ and $T_{uf} = Q_3 + 1.5 * IQR$, respectively. Similarly, $Q_1^{'}$, $Q_3^{'}$, $IQR^{'}$, $T_{lf}^{'}$, and $T_{hf}^{'}$ can be computed from the embedding array $\bar{t}_{n,g}$. Then, based on $\mathcal{L}_{l-pull}$ and $\mathcal{L}_{h-pull}$, we define two focal pulling losses as

$$\begin{aligned} &\mathcal{L}_{l-focal} = \\ &\begin{cases} \mathcal{L}_{l-pull}, \text{ if } T_{lf} < t_k\left(p_{n,g,k}\right) < T_{uf} \\ \dfrac{1}{N} \displaystyle\sum_{n=1}^{N} \sum_{g=1}^{G} \sum_{k=1}^{K_g} \alpha_{n,g,k} \left(\bar{t}_{n,g} - t_k\left(p_{n,g,k}\right)\right)^2, \text{ else,} \end{cases} \end{aligned} \tag{15}$$

and

$$\begin{aligned} &\mathcal{L}_{h-focal} = \\ &\begin{cases} \mathcal{L}_{h-pull}, \text{ if } T_{lf}^{'} < \bar{t}_{n,g} < T_{hf}^{'} \\ \dfrac{1}{N} \displaystyle\sum_{n=1}^{N} \sum_{g=1}^{G} \alpha_{n,g} \left(\bar{t}_n - \bar{t}_{n,g}\right)^2, \text{ else,} \end{cases} \end{aligned} \tag{16}$$

where α_{ngk} and α_{ng} are adjustable parameters. Finally, the hierarchical associative loss is rewritten as

$$\mathcal{L}_{hi} = \mathcal{L}_{push} + \mathcal{L}_{l-focal} + \lambda \mathcal{L}_{h-focal}. \tag{17}$$

E. Progressive Associative Decoding

In terms of progressive associative decoding, we first use standard location decoding to determine the keypoint locations

Algorithm 1 Progressive Associative Decoding

Input: The predicted tagmap t_k and the corresponding predicted keypoint coordinates $p_{n,k}$
Output: Target human poses $\mathcal{P}$
Initialize: empty pool of human instance $Ins = \{\}$, a set of keypoints kpt_{set}

for $Iteration = 1$ to K **do**
  if $len(Ins) = 0$ **then**
    for kpt in kpt_{set} **do**
      for $n = 1$ to N **do**
        Add the keypoint to $Ins.add(p_{n,1})$;
        Calculate the reference embedding $\bar{t}_n = t_1\,(p_{n,1})$;
    Remove the keypoint from $kpt_{set}.remove(1)$
  else
    for kpt in kpt_{set} **do**
      for $n = 1$ to N **do**
        Calculate the distance matrix $\mathcal{M}(t_{n,k}, \bar{t}_n)$ with (18);
        Find the keypoint with the smallest distance $k = argmin(\mathcal{M})$;
        Add the keypoint to $Ins.add(p_{n,k})$;
        Update reference embeddings with (6);
    Remove the keypoint from $kpt_{set}.remove(k)$
Return $\mathcal{P} = \{Ins\}_{n=1}^{N}$

based on the predicted heatmaps and retrieve their corresponding tag values based on the predicted tagmaps, as shown in Fig. 3. Then, keypoint grouping is implemented by comparing the tag values of keypoints in a sequential way illustrated in Fig. 5 (b).

Given the first two types of keypoints, for each pair, if their tag values fall within a specific threshold, the keypoints are associated with one human instance and added to the empty pool of human instance $Ins = \{\}$. Otherwise, the keypoints would be regarded as two distinct human instances added to Ins. Then, the corresponding reference embedding of each human instance is calculated accordingly. Starting with the third type of keypoints, their tags are compared with the reference embeddings in Ins. For each keypoint, if its tag value falls within a specific threshold compared with an existing human instance, then it is added to one certain instance. Otherwise, it is used to start a new human instance added to Ins. Afterward, the reference embeddings in Ins are updated accordingly. The above associative iteration repeats for each type of keypoints till all detections have been associated with a human instance.

One important factor in the above associative decoding is the order of keypoint types for grouping. Classical associative decoding usually provides a predefined order of updating the reference embeddings, which may affect the grouping performance especially when the tag values of two sequential types of keypoints are relatively large. Therefore, in our proposed progressive associative decoding, at each associative iteration, only the keypoint type with the strongest association to the current reference embeddings is selected. This way, the update of reference embeddings becomes much more smooth, benefiting the following keypoints for grouping. Specifically, we first initialize a set of keypoints $kpt_{set} = \{1, 2, \cdots, K\}$ to hold the remaining keypoints that need to be associated. For the first type of keypoints, the tag values are regarded as the initial reference embedding of human instances. Starting with the second type of joints, we calculate the distance matrix between the tags of all the keypoints in kpt_{set} and the reference embeddings as

$$\mathcal{M}(t_{nk},\bar{t}_n) = \begin{bmatrix} |t_{11}-\bar{t}_1| & |t_{12}-\bar{t}_1| & \cdots & |t_{1k}-\bar{t}_1| \\ |t_{21}-\bar{t}_2| & |t_{22}-\bar{t}_2| & \cdots & |t_{2k}-\bar{t}_2| \\ \vdots & \vdots & \ddots & \vdots \\ |t_{n1}-\bar{t}_n| & |t_{n2}-\bar{t}_n| & \cdots & |t_{nk}-\bar{t}_n| \end{bmatrix}. \quad (18)$$

Then, the keypoints with the smallest distance in the distance matrix are selected as the next type of keypoints for grouping and updating the reference embeddings. The method for obtaining the optimal match of keypoints based on the distance matrix is implemented by the Hungarian algorithm [50]. The pseudo code of progressive associative decoding is summarized in Algorithm 1.

IV. Experiments

We provide in this section extensive quantitative and qualitative comparison results on three publicly-available datasets MS-COCO [57], CrowdPose [54], and MPII [56] for evaluation.

A. Datasets and Evaluation Metrics

MS-COCO The MS-COCO dataset [57] is a large-scale keypoint detection benchmark containing more than 200k images and 250k human instances with 17 keypoints annotations. The train2017 set including 57k images and 150k human instances and the test-dev2017 set including 20k images and 80k human instances are utilized for training and testing respectively. OKS-based average precision (AP) and average recall (AR) are used as the official evaluation metrics. Object

TABLE I
HIERARCHICAL ASSOCIATIVE MODEL FOR THE MAINSTREAM POSE ESTIMATION DATASETS. THE INDEXES OUTSIDE OF BRACKETS CORRESPOND TO THE SERIAL NUMBERS OF KEYPOINTS IN FIG. 4(A), WITH L FOR LEFT AND R FOR RIGHT. THE INDEXES IN BRACKETS CORRESPOND TO THE ANNOTATION NUMBERS IN EACH DATASET. FLIC DEFINES THE HUMAN POSE WITH 11 KEYPOINTS. LSP, AIC, CROWDPOSE, AND HIEVE CONTAIN 14 ANNOTATED KEYPOINTS. IN MPII, EACH HUMAN INSTANCE HAS 16 ANNOTATED KEYPOINTS. IN MS-COCO AND OCHUMAN, EACH PERSON IS LABELED WITH 17 KEYPOINTS.

Dataset	Group No.									
	1	2	3	4	5	6	7	8	9	10
FLIC [51]	2(**6**), 3L(**8**), 3R(**7**), 8L(**3**), 8R(**2**)	9L(**9**), 9R(**10**)	10L(**5**)	10R(**0**)	11L(**4**)	11R(**1**)	-	-	-	-
LSP [52]	1(**13**), 5(**12**), 8L(**9**), 8R(**8**)	9L(**3**), 9R(**2**)	10L(**11**)	10R(**6**)	11L(**10**)	11R(**7**)	12L(**4**)	12R(**1**)	13L(**5**)	13R(**0**)
AIC [53]	1(**12**), 5(**13**), 8L(**0**), 8R(**3**)	9L(**6**), 9R(**9**)	10L(**2**)	10R(**5**)	11L(**1**)	11R(**4**)	12L(**7**)	12R(**10**)	13L(**8**)	13R(**11**)
CrowdPose [54]	1(**12**), 5(**13**), 8L(**1**), 8R(**0**)	9L(**7**), 9R(**6**)	10L(**5**)	10R(**4**)	11L(**3**)	11R(**2**)	12L(**9**)	12R(**8**)	13L(**11**)	13R(**10**)
HiEve [55]	2(**0**), 6(**1**), 8L(**5**), 8R(**2**)	9L(**11**), 9R(**8**)	10L(**7**)	10R(**4**)	11L(**6**)	11R(**3**)	12L(**12**)	12R(**9**)	13L(**13**)	13R(**10**)
MPII [56]	1(**9**), 5(**8**), 6(**7**), 8L(**13**), 8R(**12**)	7(**6**), 9L(**3**), 9R(**2**)	10L(**15**)	10R(**10**)	11L(**14**)	11R(**11**)	12L(**4**)	12R(**1**)	13L(**5**)	13R(**0**)
MS-COCO [57] & OCHuman [58]	2(**0**), 3L(**1**), 3R(**2**), 4L(**3**), 4R(**4**), 8L(**5**), 8R(**6**)	9L(**11**), 9R(**12**)	10L(**9**)	10R(**10**)	11L(**7**)	11R(**8**)	12L(**13**)	12R(14)	13L(**15**)	13R(**16**)

TABLE II
QUANTITATIVE COMPARISON RESULTS ON THE MS-COCO TEST-DEV2017 DATASET.

Associative Method	Location Method	Backbone	#Params	AP	AP50	AP75	APM	APL	AR	ARM	ARL
Limb-based	OpenPose [16]	CPM	25.9M	61.8	84.9	67.5	57.1	68.2	66.5	60.6	74.6
	PersonLab [28]	ResNet-152	68.7M	66.5	88.0	72.6	62.4	72.3	71.0	-	-
	KHGF [17]	Hourglass-104	187.7M	67.7	-	73.1	**65.9**	70.5	71.7	-	-
Center-based	CenterNet [22]	DLA-34	21.0M	57.9	84.7	63.1	52.5	67.4	-	-	-
	CenterNet [22]	Hourglass-104	187.7M	63.0	86.8	69.6	58.9	70.4	-	-	-
	Centripetal Offsets [59]	HRNet-W32	33.8M	66.2	87.3	71.9	62.3	74.8	70.5	63.7	79.8
	DEKR [60]	HRNet-W32	29.6M	67.3	87.9	74.1	61.5	**76.1**	72.4	65.4	81.9
	CenterGroup [61]	HrHRNet-W32	30.3M	67.6	88.7	73.6	61.9	75.6	-	-	-
Embedding-based	AE [15]	HRNet-W32	28.5M	64.1	86.3	70.4	57.4	73.9	-	-	-
	AE [15]	HrHRNet-W32	28.6M	66.4	87.5	72.8	61.2	74.2	-	-	-
Ours	AE [15]	HRNet-W32	28.5M	64.6	87.9	71.4	58.1	73.9	71.5	63.0	82.8
	AE [15]	HrHRNet-W32	28.6M	**67.8**	**89.2**	**74.8**	62.6	75.4	**73.6**	**66.6**	**83.0**

keypoint similarity (OKS) serves as a similarity measure between the ground-truth keypoints and the predicted keypoints, which can be regarded as the same as the intersection over union (IoU) in object detection, defined as

$$\mathrm{OKS} = \frac{\sum_i \exp(-d_i^2/2s^2k_i^2)\delta(v_i > 0)}{\sum_i \delta(v_i > 0)}, \tag{19}$$

where d_i is the Euclidean distance between each predicted keypoint and its relevant ground truth, v_i stands for the visibility flag of the ground truth, s denotes the object scale, and k_i is a constant that controls falloff. In addition, AP (the means of AP scores at $\mathrm{OKS} = 0.50, 0.55, ..., 0.90, 0.95$), AP^{50} (AP at $\mathrm{OKS} = 0.50$), and AP^{75} (AP at $\mathrm{OKS} = 0.75$) are adopted as standard metrics. For medium and large scale persons in the MS-COCO dataset, AP^M and AP^L are defined accordingly.

CrowdPose The CrowdPose dataset [54] is a challenging benchmark targeting human pose estimation from crowded scenarios. The dataset consists of 20k images and a total of 80k human instances annotated with 14 keypoints. The training, validation, and test sets are split according to the ratio of 5:1:4. In addition to the evaluation metrics provided by MS-COCO, extra AP scores AP^E, AP^M, and AP^H are used in the CrowdPose dataset for easy, medium, and hard instances defined by the Crowd Index.

MPII The MPII human pose multi-person dataset [56] contains 5,602 groups of images of multiple interacting people, which are split into 3,844 for training and 1,758 for testing. Each person is annotated with 16 body joints. We also use the official AP scores for the evaluation of this dataset.

As keypoint annotations may vary across different datasets, the hierarchical associative models represented in the MS-COCO, CrowdPose, MPII, and other mainstream pose estimation datasets are provided in Table I. Definitions of the corresponding keypoints can be retrieved in Fig. 4(a).

B. Implementation Details

For COCO and CrowdPose datasets, we implemented our network using an Adam optimizer with an initial learning rate of 1.5e-3 and a batch size of 32. The learning rate was decayed to 1.5e-4 and 1.5e-5 after 200 and 260 epochs. During training, we cropped and resized the training images to 512×512 and used random rotation ([-30, 30]), random scale ([0.75, 1.5]), random translation ([-40, 40]). During inference, we first resized the shorter dimension of each image to 512 while maintaining the aspect ratio, and then averaged the heatmaps

TABLE III
QUANTITATIVE COMPARISON RESULTS ON THE MS-COCO VAL2017 DATASET.

Associative Method	Location Method	Backbone	#Params	AP	AP50	AP75	APM	APL	AR	ARM	ARL
Limb-based	OpenPose [16]	CPM	25.9M	61.0	84.9	67.5	56.3	69.3	-	-	-
	PersonLab [28]	ResNet-152	68.7M	66.5	86.2	71.9	62.3	73.2	70.7	65.6	77.9
Center-based	CenterNet [22]	DLA-34	21.0M	58.9	-	-	-	-	-	-	-
	CenterNet [22]	Hourglass-104	187.7M	64.0	85.6	70.2	59.4	72.1	-	-	-
	AdaptivePose [62]	DLA-34	21.0M	64.9	86.4	70.9	58.6	74.2	70.5	-	-
	Centripetal Offsets [59]	HRNet-W32	33.8M	67.1	-	-	**62.7**	76.4	-	-	-
	DEKR [60]	HRNet-W32	29.6M	68.0	86.7	74.5	62.1	77.7	73.0	66.2	82.7
	CenterGroup [61]	HrHRNet-W32	30.3M	68.6	87.6	74.1	62.0	78.0	-	-	-
Embedding-based	AE [15]	HRNet-W32	28.5M	64.4	-	-	57.1	75.6	-	-	-
	AE [15]	HrHRNet-W32	28.6M	67.1	86.2	73.0	61.5	76.1	-	-	-
Ours	AE [15]	HRNet-W32	28.5M	64.9	86.5	70.5	55.5	69.5	71.6	62.8	83.7
	AE [15]	HrHRNet-W32	28.6M	**68.8**	**88.4**	**74.6**	62.2	**78.1**	**74.3**	**67.4**	**83.8**

TABLE IV
QUANTITATIVE COMPARISON RESULTS ON THE CROWDPOSE TESTING DATASET.

Associative Method	Location Method	Backbone	#Params	AP	AP50	AP75	APE	APM	APH
-	Mask R-CNN [63]	Faster R-CNN	-	57.2	83.5	60.3	69.4	57.9	42.0
	SimpleBaseline [4]	ResNet-152	68.6M	60.8	81.4	65.7	71.4	61.2	51.2
	RMPE [13]	4-stack Hourglass	-	61.0	81.3	66.0	71.2	61.4	51.1
Limb-based	Openpose [16]	CPM	25.9M	-	-	-	62.7	48.7	32.3
	KHGF [17]	Hourglass-104	187.7M	65.3	85.9	69.6	74.1	66.3	55.1
	KHGF [17]	Hourglass-104MA	226.9M	67.3	87.2	71.2	**76.4**	68.2	56.9
Center-based	DEKR [60]	HRNet-W32	29.6M	65.7	85.7	70.4	73.0	66.4	57.5
	Centripetal Offsets [59]	HRNet-W32	33.8M	66.6	-	-	75.5	67.7	56.3
	CenterGroup [61]	HrHRNet-W48	65.5M	67.6	87.7	72.7	73.9	68.2	**60.3**
Embedding-based	AE [15]	HrHRNet-W32	28.6M	64.2	85.5	68.7	72.0	64.7	56.2
	AE [15]	HrHRNet-W48	63.8M	65.9	86.4	70.6	73.3	66.5	57.9
Ours	AE [15]	HrHRNet-W32	28.6M	**67.6**	**87.9**	**73.2**	75.4	**68.3**	59.9

TABLE V
QUANTITATIVE COMPARISON RESULTS ON THE MPII TESTING DATASET.

Method	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	Total
DeeperCut [20]	78.4	72.5	60.2	51.0	57.2	52.0	45.4	59.5
OpenPose [16]	91.2	87.6	77.7	66.8	75.4	68.9	61.7	75.6
RMPE [13]	88.4	86.5	78.6	70.4	74.4	73.0	65.8	76.7
AE [15]	92.1	89.3	78.9	69.8	76.2	71.6	64.7	77.5
SPM [38]	89.7	87.4	80.4	72.4	76.7	74.9	68.3	78.5
Ours	**93.5**	**90.6**	**82.6**	**72.8**	**79.4**	**75.0**	**68.3**	**80.3**

of both the original and the flipped images to compute the final heatmaps.

For the MPII dataset, the network was trained with an initial learning rate of 3e-3 and decreased to 1.5e-3/7.5e-4/3e-4/1.5e4 at the 100th/150th/170th/200th epoch. During training, we augmented each sample with random rotation ([-40, 40]), random scale ([0.7, 1.3]), and the same random translation as COCO and CrowdPose. During inference, we cropped image patches using the given position and averaged the person scale of test images, and resized and padded the samples to 384×384 as input to HAE.

All methods were implemented within the PyTorch framework and trained on two NVIDIA Tesla A100 40GB GPUs for 300 epochs.

C. Quantitative Analysis

Quantitative results compared to the state-of-the-art methods on the MS-COCO test-dev 2017 dataset are stated in Table II. Among all bottom-up approaches, the proposed HAE achieves the best performance and outperforms the state-of-the-art embedding method AE [15] by 1.4% in AP. Compared with the state-of-the-art limb-based KHGF [17] and center-based CenterGroup [61], we still achieve a slight performance improvement with fewer parameters (187.7M to 28.6M). More importantly, HAE and PAD achieve consistent performance improvements regardless of the location methods and the backbones, proving its architecture-agnostic effectiveness. Similar results can be observed on the MS-COCO val2017 set as provided in Table III.

Quantitative results on the CrowdPose dataset, including three crowding levels, are summarized in Table IV. For a fair comparison with embedding-based methods, we

TABLE VI
ABLATION STUDIES OF THE HYPER-PARAMETER λ ON THE CROWDPOSE TESTING DATASET. BASELINE IS THE METHOD WITHOUT HAE.

λ	AP	AP^{50}	AP^{75}	AP^{E}	AP^{M}	AP^{H}
Baseline	64.2	85.5	68.7	72.0	64.7	56.2
1.5	**66.1**	**86.5**	**70.7**	**73.5**	**66.6**	**58.3**
2.0	65.8	86.5	70.5	73.3	66.3	58.0
3.0	65.0	85.6	69.4	72.7	65.5	57.3

TABLE VII
ABLATION STUDIES OF THE ASSOCIATIVE MODEL ON THE CROWDPOSE TESTING DATASET. SP DENOTES THE HIERARCHICAL MODEL BASED ON SPATIAL PRIOR IN FIG. 4(D), AND MI DENOTES THE HIERARCHICAL MODEL BASED ON MUTUAL INFORMATION IN FIG. 4(E).

Associative Model	AP	AP^{50}	AP^{75}	AP^{E}	AP^{M}	AP^{H}
Baseline	64.2	85.5	68.7	72.0	64.7	56.2
SP	65.6	86.3	70.4	72.9	66.1	57.6
MI	65.7	86.4	70.4	73.4	66.2	58.1
HAE	**66.1**	**86.5**	**70.7**	**73.5**	**66.6**	**58.3**

TABLE VIII
ABLATION STUDIES OF THE FOCAL PULLING LOSS ON THE CROWDPOSE TESTING DATASET.

Method	AP	AP^{50}	AP^{75}	AP^{E}	AP^{M}	AP^{H}
Baseline	64.2	85.5	68.7	72.0	64.7	56.2
OHKM	65.5	85.8	69.8	73.4	66.0	57.7
IQR	**65.9**	**86.5**	**70.7**	**73.4**	**66.3**	**58.5**

TABLE IX
ABLATION STUDIES OF THE PROGRESSIVE ASSOCIATIVE DECODING ON THE CROWDPOSE TESTING DATASET. BASELINE IS THE METHOD WITH HAE AND STANDARD ASSOCIATIVE DECODING, TDAD DENOTES TOP-DOWN ASSOCIATIVE DECODING, AND BUAD DENOTES BOTTOM-UP ASSOCIATIVE DECODING.

Method	AP	AP^{50}	AP^{75}	AP^{E}	AP^{M}	AP^{H}
Baseline	67.4	87.8	72.6	74.7	67.8	59.7
TDAD	67.4	87.9	72.6	74.6	67.8	59.6
BUAD	66.8	87.1	71.8	74.5	67.3	58.9
PAD	**67.6**	**87.9**	**73.2**	**75.4**	**68.3**	**59.9**

TABLE X
RUNTIME ANALYSIS OF DIFFERENT ASSOCIATIVE DECODING. NMS IS SHORT FOR NON-MAXIMUM SUPPRESSION, MATCH DENOTES GROUP KEYPOINTS TO HUMAN POSES, ADJUST DENOTES 0.25 SUB-PIXEL ADJUSTMENT WITH (5), AND REFINE IDENTIFIES MISSING KEYPOINTS.

Time [ms/img]	NMS	Match	Adjust	Refine	Total
Baseline	6.1	**7.6**	8.2	2584.8	2606.7
PAD	**6.1**	25.0	**7.8**	**2535.3**	**2574.2**

re-implemented AE [15] on the CrowdPose dataset with HrHRNet-W32 as the backbone. As the CrowdPose dataset contains more complex poses and occlusions, the hierarchical associative encoding and decoding approach would bring more performance improvements compared to the MS-COCO dataset. According to Table IV, bottom-up approaches generally outperform top-down approaches especially on dealing with hard samples. It is because human instances of the CrowdPose dataset can be heavily overlapped, leading to poor person detection and erroneous keypoints for pose estimation. Therefore, our proposed method outperforms the best top-down approach RMPE [13] by a large margin of 6.6% in AP. Compared to AE [15], our proposed approach achieves an average increase of 3.4% in AP with the same HrHRNet-W32 backbone, which is much more significant compared to the MS-COCO test-dev2017 dataset. Even adopting a more powerful backbone HrHRNet-W48, AE [15] still fails to approach the proposed method, with an average decrease of 1.7% in AP. In addition, we achieve comparable results among all the state-of-the-art methods listed with fewer model parameters on the CrowdPose dataset. Table V reports results on the MPII dataset. Our proposed framework achieves 80.3% in terms of the total AP score, outperforming all the previous bottom-up methods. In addition, HAE could bring noticeable improvement over AE, especially on the elbow and ankle (+3.7% and +3.6%), confirming its effectiveness on those hard-to-associate keypoints.

D. Ablation Study

To determine the individual contribution of associative encoding and decoding components, we conduct ablation studies on the CrowdPose dataset. For a fair comparison, HrHRNet-W32 [39] is selected as the backbone network and the standard location/associative encoding and decoding scheme in Section III.B is defined as the baseline.

Hierarchical associative loss The hyper-parameter λ in the hierarchical associative loss in (17) is to balance the low-level and the high-level pulling losses. Setting a larger λ would force the model to focus more on the group-level association. To validate this, experimental results of both the baseline and the hierarchical associative loss with different λ on the CrowdPose dataset are reported in Table VI. Setting $\lambda = 1.5$ achieves the best performance, leading to an average increase of 1.9% in AP compared to the baseline. Along with the increase of λ (*i.e.*, from 1.5 to 3.0), the network would relatively ignore the keypoint-level association, which in turn degrades the overall performance.

Hierarchical associative model Hierarchical models (such as Figs. 4(d) and (e)) are designed to improve localization performance, and association performance improvement may not be satisfactory when introduced to associative algorithms. To demonstrate the effectiveness of our designed hierarchical associative model (Fig. 4(c)), comparison results of different associative models are reported in Table VII. Though hierarchical models yield significant improvements, HAE is proven to be the most effective one.

Focal pulling loss In addition to the IQR-based method to penalize the hard-to-associate keypoints, OHKM can also be applied to mine the hard keypoints which punishes the top M $(M < K)$ keypoint losses. To evaluate the effectiveness of the proposed focal pulling loss, both OHKM-based and IQR-based methods are implemented onto the same baseline for comparison as stated in Table VIII. Compared to the baseline, adopting the OHKM-based method achieves

This article has been accepted for publication in IEEE Transactions on Circuits and Systems for Video Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2022.3215564

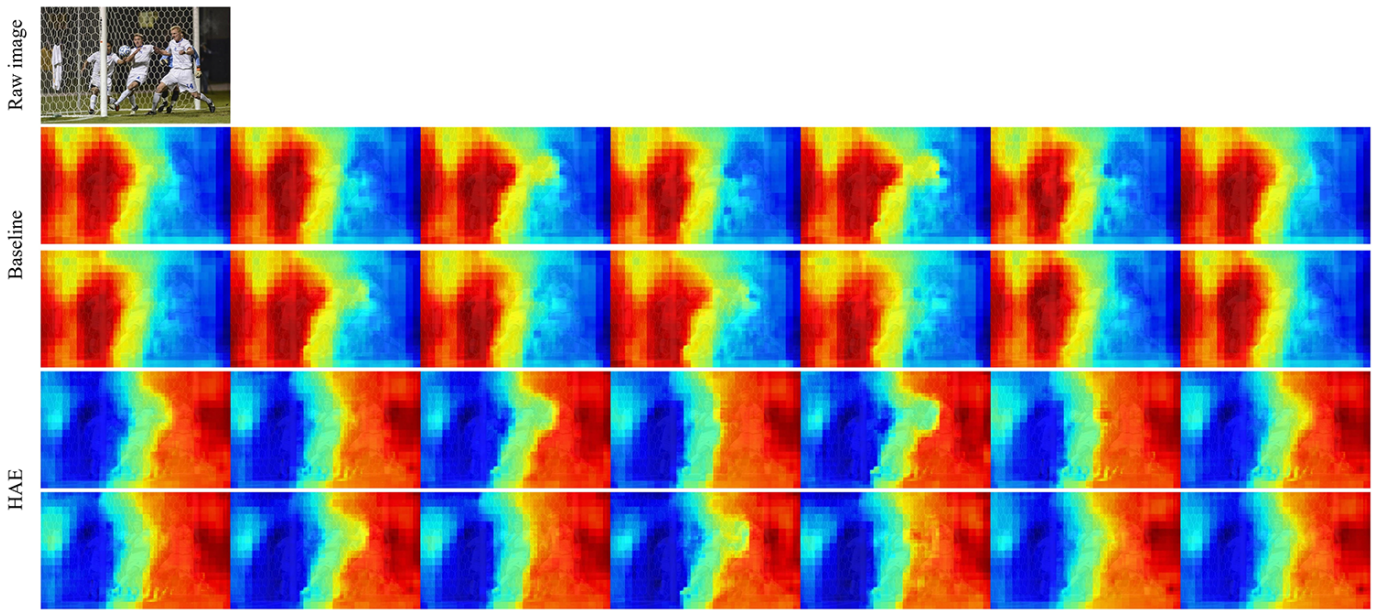

Fig. 7. Exemplar tagmaps produced with and without hierarchical associative encoding (HAE).

Fig. 8. Qualitative human pose estimation results on the CrowdPose dataset. Rows 1-2: The final poses inferred by the standard associative decoding and the proposed progressive associative decoding respectively. Highlight regions are marked by white circles.

TABLE XI
ABLATION STUDIES OF DIFFERENT COMPONENT COMBINATIONS OF THE PROPOSED FRAMEWORK ON THE CROWDPOSE TESTING DATASET, INCLUDING HIERARCHICAL ASSOCIATIVE LOSS (HAL), FOCAL PULLING LOSS (FPL), AND PROGRESSIVE ASSOCIATIVE DECODING (PAD).

HAL	FPL	PAD	AP	AP^{50}	AP^{75}	AR	AR^{50}	AR^{75}	AP^{E}	AP^{M}	AP^{H}
			64.2	85.5	68.7	70.7	89.3	74.7	72.0	64.7	56.2
✓			66.1	86.5	70.7	72.6	90.5	76.8	73.5	66.6	58.3
✓	✓		67.4	87.8	72.6	74.6	92.1	79.2	74.7	67.8	59.7
✓	✓	✓	**67.6**	**87.9**	**73.2**	**75.7**	**93.1**	**80.4**	**75.4**	**68.3**	**59.9**

an average increase of 1.3% in AP while the AP increase achieved by the IQR-based method is 1.7%. Though limited, the performance gap between OHKM and IQR in Table VIII validates the effectiveness of IQR in distinguishing hard-to-associate samples.

Progressive associative decoding To better validate the effectiveness of PAD, we set two additional predefined orders of updating the reference embeddings. Top-down associative decoding updates the reference embeddings head-to-toe and bottom-up in the opposite order. Taking the CrowdPose dataset as an example, top-down associative decoding TD = [12,13,0,1,2,3,4,5,6,7,8,9,10,11], bottom-up associative decoding BU = [11,10,9,8,7,6,5,4,3,2,1,0,13,12], while baseline BASE = [0,1,2,3,4,5,6,7,8,9,10,11,12,13] according to Fig. 4

Fig. 9. Qualitative human pose estimation results on the UAV-Human dataset.

(a) and Table I. As shown in Table IX, an unreasonable update order (BUAD) decreases AP by 0.6% in the inference process, while the proposed PAD is able to adaptively choose the best order to make the update smoother and achieve the best performance.

Runtime analysis Runtime statistics of each component in standard associative decoding and PAD are summarized in Table X. Specifically, we measure the average time during the inference of 1000 images to obtain stable results. All methods were tested on the same machine with one NVIDIA RTX 2080Ti 11GB GPU. The results suggest that PAD even shortens the total inference time. It is because the time cost added by PAD during the matching stage is almost negligible and a reasonable association is beneficial for cost reduction during the refinement stage.

Effectiveness of each component To quantify the contribution of each component in the proposed associative encoding and decoding methods, ablation studies of different component combinations on the CrowdPose testing dataset are conducted as stated in Table XI. Among all components, deploying the hierarchical associative loss obtains the most performance improvements, leading to an average increase of 1.9% in AP. Then, introducing the IQR-based focal pulling loss to penalize more on the hard-to-associate samples further improves the AP performance by 1.3%. Though limited, the 0.2% increase in AP demonstrates that PAD is a simple yet effective step for better inference without introducing additional computational costs. The idea of adaptively selecting the best inference order is extendable to other pose estimation frameworks (such as limb- or center-based methods) for performance improvement.

E. Qualitative Analysis

One main contribution of our proposed approach is introducing group-level associative encoding to complement keypoint-level and the instance-level associations. To validate the effectiveness of hierarchical associative encoding, we visualize the tagmap results as shown in Fig. 7, where different tag values are assigned with different colors. As described in Section III.C, the more distinguishable the tag values are, the more beneficial they are for keypoint grouping and human pose estimation. Tag values in embedding-based methods are trained in an unsupervised manner, where their exact values

(a) (b) (c) (d)

Fig. 10. Some representative failure cases: (a) unreasonable keypoint associations across two persons in crowded scenes, (b) missed keypoint associations caused by undetected keypoints, and (c-d) false positive associations.

are meaningless as long as they are separable. As a result, tag values of the same input human instances can be different in different approaches as shown in Fig. 7. Compared to the baseline tagmaps, the boundaries in the tagmaps produced by HAE are much "clearer". As a result, HAE can better distinguish different human instances, which is consistent with the quantitative results in Table XI (*i.e.* 1.9% increase in AP).

Exemplar human pose estimation results on the CrowdPose dataset are shown in Fig. 8. Following standard associative decoding, though the majority of keypoints are effectively grouped, we still observe a considerable number of isolated keypoints. Comparatively, PAD achieves better association results, and these isolated keypoints are reasonably associated. As the isolated keypoints only cover a small ratio, recovering them through PAD would not bring significant quantitative performance improvements. It explains why the increase in AP in Table XI brought by PAD is relatively limited. In addition, we directly apply the pre-trained on the CrowdPose dataset to the UAV-Human dataset [64] of single-person and multi-person scenes, which has 22,476 annotated frames for pose estimation. Qualitative results in Fig. 9 confirm the overall good performance of our proposed method. Typical failure cases about keypoint grouping are summarized in Fig. 10. As seen, unreasonable keypoint associations (Fig. 10(a)) may occur in mutual occlusion, and our approach well addresses it with the prior knowledge of human skeletons. For other failure cases, such as false positive (Fig. 10(b)) or false negative (Fig. 10(c-d)) associations, it cannot be perfectly solved by just improving the association algorithm, as it relies on correct keypoints coordinates. Hence, associative encoding and decoding should be jointly considered with location encoding and decoding for further performance improvement.

V. Conclusion

In this paper, we have proposed a hierarchical associative model based on human anatomy. For training, we constructed a hierarchical associative loss to pursue better keypoint-, group-, and instance-level associations. In addition, to encourage the network to focus more on the hard-to-associate keypoints, a focal pulling loss was designed to increase the weights of those keypoints. For inference, we proposed a progressive associative decoding scheme that follows a better order for keypoint grouping and pose estimation. Quantitive and qualitative results on both the MS-COCO and the CrowdPose datasets demonstrated the effectiveness of our proposed approach, especially in dealing with heavily occluded human poses. Considering the architecture independence of hierarchical associative encoding and decoding, we believe that our work

will lead to more research efforts on bottom-up human pose estimation.

REFERENCES

[1] Z. Yang, Y. Li, J. Yang, and J. Luo, "Action recognition with spatio-temporal visual attention on skeleton image sequences," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 8, pp. 2405–2415, 2018.

[2] H. Wu, X. Ma, and Y. Li, "Spatiotemporal multimodal learning with 3d cnns for video action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 3, pp. 1250–1261, 2021.

[3] H. Zheng and X. Zhang, "A cross view learning approach for skeleton-based action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 5, pp. 3061–3072, 2021.

[4] B. Xiao, H. Wu, and Y. Wei, "Simple baselines for human pose estimation and tracking," in *Proc. ECCV*, 2018, pp. 466–481.

[5] S. You, H. Yao, and C. Xu, "Multi-target multi-camera tracking with optical-based pose association," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 8, pp. 3105–3117, 2020.

[6] X. Sun, J. Zhou, W. Zhang, Z. Wang, and Q. Yu, "Robust monocular pose tracking of less-distinct objects based on contour-part model," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 11, pp. 4409–4421, 2021.

[7] Z. Zheng, L. Zheng, and Y. Yang, "Pedestrian alignment network for large-scale person re-identification," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 10, pp. 3037–3045, 2018.

[8] J. Lei, L. Niu, H. Fu, B. Peng, Q. Huang, and C. Hou, "Person re-identification by semantic region representation and topology constraint," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 8, pp. 2453–2466, 2018.

[9] M. Ye, J. Shen, G. Lin, T. Xiang, L. Shao, and S. C. Hoi, "Deep learning for person re-identification: A survey and outlook," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 6, pp. 2872–2893, 2021.

[10] L. Ma, X. Jia, Q. Sun, B. Schiele, T. Tuytelaars, and L. Van Gool, "Pose guided person image generation," *Proc. NeurIPS*, vol. 30, 2017.

[11] B. Chen, Y. Zhang, H. Tan, B. Yin, and X. Liu, "Pman: Progressive multi-attention network for human pose transfer," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 1, pp. 302–314, 2021.

[12] W. Wang, Y. Xu, J. Shen, and S.-C. Zhu, "Attentive fashion grammar network for fashion landmark detection and clothing category classification," in *Proc. CVPR*, 2018, pp. 4271–4280.

[13] H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu, "Rmpe: Regional multi-person pose estimation," in *Proc. ICCV*, 2017, pp. 2334–2343.

[14] L. Zhao, J. Xu, C. Gong, J. Yang, W. Zuo, and X. Gao, "Learning to acquire the quality of human pose estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 4, pp. 1555–1568, 2020.

[15] A. Newell, Z. Huang, and J. Deng, "Associative embedding: End-to-end learning for joint detection and grouping," in *Proc. NeurIPS*, 2017, pp. 2277–2287.

[16] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Openpose: Realtime multi-person 2d pose estimation using part affinity fields," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 01, pp. 172–186, 2021.

[17] J. Li and M. Wang, "Multi-person pose estimation with accurate heatmap regression and greedy association," *IEEE Trans. Circuits Syst. Video Technol.*, 2022.

[18] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Proc. NeurIPS*, vol. 28, 2015.

[19] L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. V. Gehler, and B. Schiele, "Deepcut: Joint subset partition and labeling for multi person pose estimation," in *Proc. CVPR*, 2016, pp. 4929–4937.

[20] E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, and B. Schiele, "Deepercut: A deeper, stronger, and faster multi-person pose estimation model," in *Proc. ECCV*, 2016, pp. 34–50.

[21] U. Iqbal and J. Gall, "Multi-person pose estimation with local joint-to-person associations," in *Proc. ECCV*, 2016, pp. 627–642.

[22] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," *arXiv:1904.07850*, 2019.

[23] G. J. Tortora and B. H. Derrickson, *Principles of anatomy and physiology*. John Wiley & Sons, 2018.

[24] K. R. Senior *et al.*, *Bone and Muscle: Structure, Force, and Motion*. The Rosen Publishing Group, Inc, 2010.

[25] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," in *Proc. CVPR*, 2016, pp. 4724–4732.

[26] "Stacked hourglass networks for human pose estimation," in *Proc. ECCV*, 2016, pp. 483–499.

[27] G. Papandreou, T. Zhu, N. Kanazawa, A. Toshev, J. Tompson, C. Bregler, and K. Murphy, "Towards accurate multi-person pose estimation in the wild," in *Proc. CVPR*, 2017, pp. 4903–4911.

[28] G. Papandreou, T. Zhu, L.-C. Chen, S. Gidaris, J. Tompson, and K. Murphy, "Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model," in *Proc. ECCV*, 2018, pp. 269–286.

[29] C. Du, H. Yu, and L. Yu, "A scale-sensitive heatmap representation for multi-person pose estimation," *IET Image Processing*, 2022.

[30] H. Yu, C. Du, and L. Yu, "Scale-aware heatmap representation for human pose estimation," *Pattern Recognition Letters*, vol. 154, pp. 1–6, 2022.

[31] G. Wei, C. Lan, W. Zeng, and Z. Chen, "View invariant 3d human pose estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 12, pp. 4601–4610, 2019.

[32] F. Zhang, X. Zhu, H. Dai, M. Ye, and C. Zhu, "Distribution-aware coordinate representation for human pose estimation," in *Proc. CVPR*, 2020, pp. 7093–7102.

[33] C. Xie, D. Zhang, Y. Hu, and Y. Chen, "Hierarchical dynamic programming module for human pose refinement," *IEEE Trans. Circuits Syst. Video Technol.*, 2022.

[34] S. Jin, W. Liu, E. Xie, W. Wang, C. Qian, W. Ouyang, and P. Luo, "Differentiable hierarchical graph grouping for multi-person pose estimation," in *Proc. ECCV*, 2020, pp. 718–734.

[35] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Trans. Neural Networks*, vol. 20, no. 1, pp. 61–80, 2008.

[36] J. Li, W. Su, and Z. Wang, "Simple pose: Rethinking and improving a bottom-up approach for multi-person pose estimation," in *Proc. AAAI*, 2020, pp. 11 354–11 361.

[37] S. Kreiss, L. Bertoni, and A. Alahi, "Pifpaf: Composite fields for human pose estimation," in *Proc. CVPR*, 2019, pp. 11 977–11 986.

[38] X. Nie, J. Feng, J. Zhang, and S. Yan, "Single-stage multi-person pose machines," in *Proc. ICCV*, 2019, pp. 6951–6960.

[39] B. Cheng, B. Xiao, J. Wang, H. Shi, T. S. Huang, and L. Zhang, "Higherhrnet: Scale-aware representation learning for bottom-up human pose estimation," in *Proc. CVPR*, 2020, pp. 5386–5395.

[40] W. Wang, Z. Zhang, S. Qi, J. Shen, Y. Pang, and L. Shao, "Learning compositional neural information fusion for human parsing," in *Proc. ICCV*, 2019, pp. 5703–5713.

[41] W. Wang, T. Zhou, S. Qi, J. Shen, and S.-C. Zhu, "Hierarchical human semantic parsing with comprehensive part-relation modeling," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2021.

[42] H.-S. Fang, Y. Xu, W. Wang, X. Liu, and S.-C. Zhu, "Learning pose grammar to encode human body configuration for 3d pose estimation," in *Proc. AAAI*, vol. 32, no. 1, 2018.

[43] T. Zhou, W. Wang, S. Liu, Y. Yang, and L. Van Gool, "Differentiable multi-granularity human representation learning for instance-aware human semantic parsing," in *Proc. CVPR*, 2021, pp. 1622–1631.

[44] Y. Tian, C. L. Zitnick, and S. G. Narasimhan, "Exploring the spatial hierarchy of mixture models for human pose estimation," in *Proc. ECCV*, 2012, pp. 256–269.

[45] W. Tang and Y. Wu, "Does learning specific features for related parts help human pose estimation?" in *Proc. CVPR*, 2019, pp. 1107–1116.

[46] S. Park, B. X. Nie, and S.-C. Zhu, "Attribute and-or grammar for joint parsing of human pose, parts and attributes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 7, pp. 1555–1569, 2017.

[47] W. Tang, P. Yu, and Y. Wu, "Deeply learned compositional models for human pose estimation," in *Proc. ECCV*, 2018, pp. 190–206.

[48] W. Platzer, *Color atlas of human anatomy: locomotor system*. Thieme, 2009, vol. 1.

[49] Y. Chen, Z. Wang, Y. Peng, Z. Zhang, G. Yu, and J. Sun, "Cascaded pyramid network for multi-person pose estimation," in *Proc. CVPR*, 2018, pp. 7103–7112.

[50] H. W. Kuhn, "The hungarian method for the assignment problem," *Nav. Res. Logist.*, vol. 2, no. 1-2, pp. 83–97, 1955.

[51] B. Sapp and B. Taskar, "Modec: Multimodal decomposable models for human pose estimation," in *Proc. CVPR*, 2013, pp. 3674–3681.

[52] S. Johnson and M. Everingham, "Clustered pose and nonlinear appearance models for human pose estimation." in *Proc. BMVC*, vol. 2, no. 4. Citeseer, 2010, p. 5.

[53] J. Wu, H. Zheng, B. Zhao, Y. Li, B. Yan, R. Liang, W. Wang, S. Zhou, G. Lin, Y. Fu *et al.*, "Large-scale datasets for going deeper in image understanding," in *Proc. ICME*, 2019, pp. 1480–1485.

[54] J. Li, C. Wang, H. Zhu, Y. Mao, H.-S. Fang, and C. Lu, "Crowdpose: Efficient crowded scenes pose estimation and a new benchmark," in *Proc. CVPR*, 2019, pp. 10 863–10 872.

[55] W. Lin, H. Liu, S. Liu, Y. Li, R. Qian, T. Wang, N. Xu, H. Xiong, G.-J. Qi, and N. Sebe, "Human in events: A large-scale benchmark for human-centric video analysis in complex events," *arXiv:2005.04490*, 2020.
[56] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, "2d human pose estimation: New benchmark and state of the art analysis," in *Proc. CVPR*, 2014, pp. 3686–3693.
[57] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Proc. ECCV*, 2014, pp. 740–755.
[58] S.-H. Zhang, R. Li, X. Dong, P. Rosin, Z. Cai, X. Han, D. Yang, H. Huang, and S.-M. Hu, "Pose2seg: Detection free human instance segmentation," in *Proc. CVPR*, 2019, pp. 889–898.
[59] L. Jin, X. Wang, X. Nie, L. Liu, Y. Guo, and J. Zhao, "Grouping by center: Predicting centripetal offsets for the bottom-up human pose estimation," *IEEE Trans. Multimedia*, 2022.
[60] Z. Geng, K. Sun, B. Xiao, Z. Zhang, and J. Wang, "Bottom-up human pose estimation via disentangled keypoint regression," in *Proc. CVPR*, 2021, pp. 14 676–14 686.
[61] G. Brasó, N. Kister, and L. Leal-Taixé, "The center of attention: Center-keypoint grouping via attention for multi-person pose estimation," in *Proc. ICCV*, 2021, pp. 11 853–11 863.
[62] Y. Xiao, X. J. Wang, D. Yu, G. Wang, Q. Zhang, and H. Mingshu, "Adaptivepose: Human parts as adaptive points," in *Proc. AAAI*, vol. 36, no. 3, 2022, pp. 2813–2821.
[63] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proc. ICCV*, 2017, pp. 2961–2969.
[64] T. Li, J. Liu, W. Zhang, Y. Ni, W. Wang, and Z. Li, "Uav-human: A large benchmark for human behavior understanding with unmanned aerial vehicles," in *Proc. CVPR*, 2021, pp. 16 266–16 275.

Congju Du received the B.Eng. and M.Sc. degrees from University of Electronic Science and Technology of China, Chengdu, China, in 2016 and 2019, respectively. He is currently pursuing the Ph.D. degree at the School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan, China. His research interests mainly include computer vision, image processing, and pattern recognition.

Zengqiang Yan is an Associate Professor at Huazhong University of Science and Technology, China. He recieved the Ph.D. degree in the Department of Computer Science and Engineering of Hong Kong University of Science and Technology, Hong Kong, in 2020. His research interests including artificial intelligence, computer vision, and medical image analysis.

Han Yu received the B.Eng. and M.Sc. degrees in electronic information engineering from HuaZhong University of Science and Technology, Wuhan, China, in 2019 and 2022, respectively. His research interests mainly include human related computer vision tasks, such as human pose estimation and human action recognition.

Li Yu received the Ph.D. degree from the Department of Electronics and Information Engineering, Huazhong University of Science and Technology, Wuhan, China in 1999 and then joined Huazhong University of Science and Technology, where she is currently a Professor with the School of Electronic Information and Communications. Her research interests include artificial intelligent, image and video processing, multimedia communications, and wireless networking.

Zixiang Xiong received his Ph.D. degree in electrical engineering from the University of Illinois at Urbana-Champaign in 1996. Since 1999, he has been with the Department of Electrical and Computer Engineering at Texas A&M University, where he is a professor and an associate department head. His main research interest lies in image/video processing, networked multimedia, and minimum-energy network communications.

Dr. Xiong received an NSF Career Award in 1999, an ARO Young Investigator Award in 2000, and an ONR Young Investigator Award in 2001. He is co-recipient of the 2006 IEEE Signal Processing Magazine best paper award, top 10% paper awards at the 2011 and 2015 IEEE Multimedia Signal Processing Workshops, an IBM best student paper award at the 2016 IEEE International Conference on Pattern Recognition, and the best demo paper award at the 2018 IEEE International Conference on Multimedia and Expo. He served as an Associate Editor for five IEEE Transactions. He was the Publications Chair of ICASSP 2007, the Technical Program Committee Co-Chair of ITW 2007, the Tutorial Chair of ISIT 2010, and the Awards Chair of Globecom 2014.