Human Pose Estimation Using Skeletal Heatmaps

Jinyoung Jun, Jae-Han Lee, and Chang-Su Kim
School of Electrical Engineering, Korea University, Seoul, Korea
E-mail: jyjun@mcl.korea.ac.kr, jaehanlee@mcl.korea.ac.kr, changsukim@korea.ac.kr

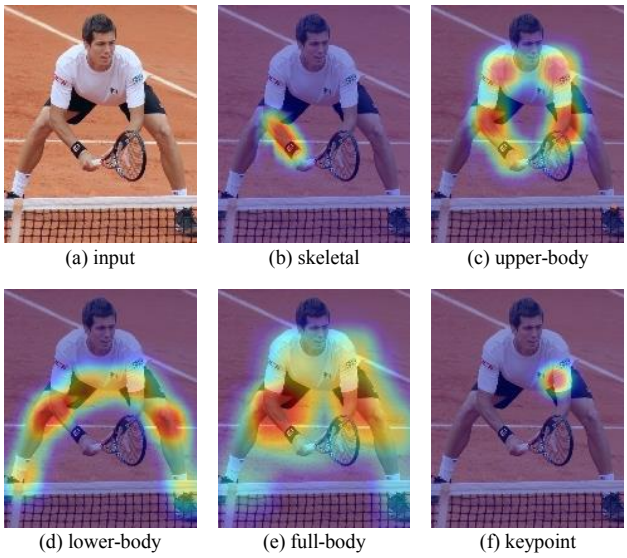
***Abstract*—We propose a novel skeletal attention module to generate keypoint heatmaps, which exploits skeletal, as well as overall body structure, information for human pose estimation. We first add augmenting convolutional layers to an existing deep neural network in order to yield skeletal heatmaps. These skeletal heatmaps emphasize keypoint relations connected either physically or virtually. By combining the skeletal heatmaps, we generate body attention maps for upper-body, lower-body, and full-body. Then, the skeletal heatmaps and the body attention maps are employed to estimate the heatmap for each keypoint. Finally, we perform weighted inference on the output heatmaps for more precise estimates. Experimental results demonstrate that the proposed algorithm enhances performance on two datasets for human pose estimation.**

I. Introduction

The objective of human pose estimation is to estimate humans' anatomical keypoint locations (*e.g.*, eyes and shoulders) in a given image. Human pose estimation is one of the fundamental problems in computer vision and can provide useful information for various applications, such as action recognition [1], segmentation [2], tracking [3] and robotic task learning [4].

Traditional approaches use pictorial structures [5], which represent a human body as a set of rigid parts and paired part connections. The pictorial structures are encoded using tree [6] or loopy [7] models. The tree model uses a belief propagation algorithm that provides a relatively fast and precise inference but may suffer from double counting. On the other hand, the loopy model uses pairwise relations between candidate locations, which may demand huge time complexity proportional to the number of candidates. It often performs approximations during inference to reduce the complexity, degrading the inference precision. Although there have been some attempts [8], [9] to overcome these shortcomings, they are susceptible to overfitting to particular datasets. Moreover, due to many challenging factors, such as occlusions, various clothing, cluttered background, and scale differences between body parts, conventional approaches do not yield reliable performance in general.

Recent advances in deep neural networks enable computers to learn inherent features from big data effectively. Thus, many attempts have been made to perform human pose estimation based on deep learning. Recent studies can be classified into two categories: regression methods [10], [11] and heatmap methods [12], [13]. A regression method estimates the locations of keypoints directly. Toshev and Szegedy [10] used cascaded CNNs to regress the spatial coordinates of body joints directly, and Carreira *et al.* [11] adopted the iterative error feedback to regress keypoint locations. A heatmap method generates a heatmap for each keypoint and selects the highest value on the heatmap as the estimated location. Newell *et al.* [12] proposed the stacked hourglass networks to capture various spatial information. Wei *et al.* [13] proposed the convolutional pose machines to increase the receptive field using multiple-stage networks. However, despite these advances, human pose estimation is still a challenging problem.



(a) input (b) skeletal (c) upper-body (d) lower-body (e) full-body (f) keypoint

Fig. 1. Given an input image in (a), the proposed algorithm generates skeletal heatmaps, one of which is shown in (b). It also generates the upper-body attention map in (c), lower-body attention map in (d), and full-body attention map in (e). Finally, the heatmap for each keypoint, such as (f), is generated. Note that the attention maps highlight keypoint locations.

Some previous algorithms [15], [16] attempt to use implicit human body structure or body parts' adjacency for multi-person pose estimation. Cao *et al.* [15] proposed part affinity fields to encode the location and orientation of limbs. Their algorithm first predicts confidence maps and affinity fields for each keypoint and then associates the keypoints with body part candidates based on the bipartite matching. Papandreou *et al.* [16] used part-induced geometric embedding to associate instance segmentation with pose estimation. Their algorithm searches all keypoints, regardless of which person they belong to. Using a tree-structured graph of each person and various range offsets predicted by convolutional neural networks, it groups the keypoints and performs the instance

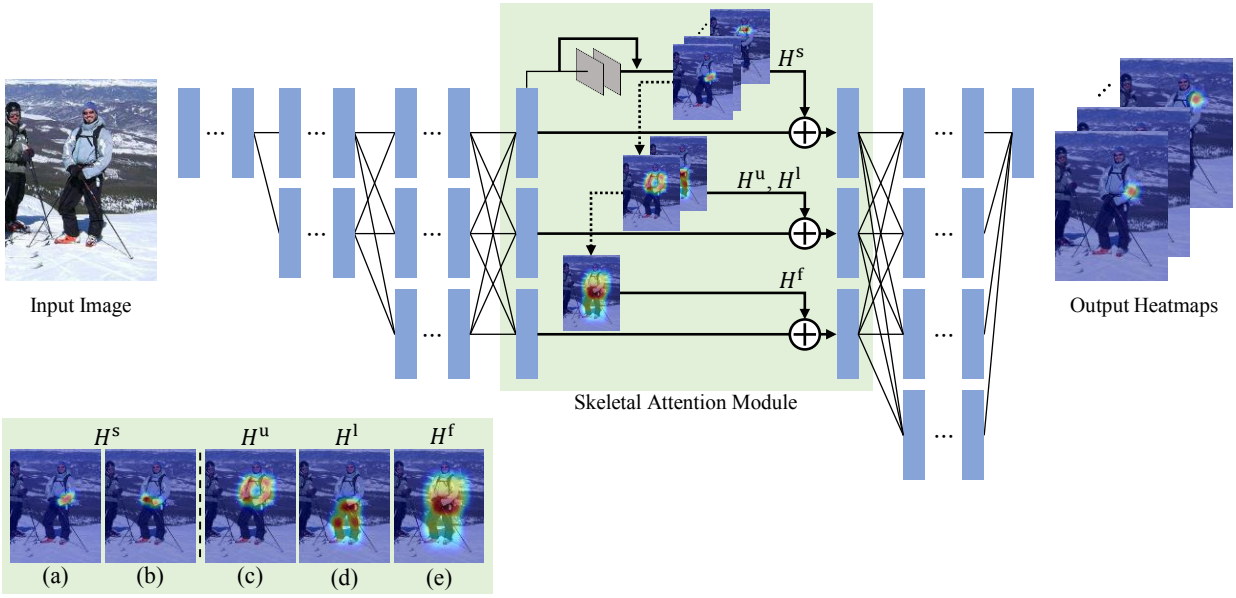


Fig. 2. An overview of the proposed algorithm. The skeletal attention module generates skeletal heatmaps H^{s}, such as (a) and (b). It then combines the skeletal heatmaps to yield the upper-body, lower-body, and full-body attention maps H^{u}(c), H^{l}(d), and H^{f}(e). The proposed algorithm applies these skeletal heatmaps and body attention maps as an additional input to the final stage of the baseline network HRNet [14] to yield the heatmap for each keypoint.

segmentation [17]. Both algorithms use structural information to identify keypoints.

In this paper, we develop a skeletal attention module to exploit full-body attention of a human instance to detect each keypoint effectively, as illustrated in Fig. 1. The skeletal attention module generates two kinds of maps, called skeletal heatmaps and body attention maps. Note that generating such auxiliary maps has been attempted in other vision tasks, including [18]. First, we add a skeletal attention module to an existing convolutional network, HRNet [14], to estimate skeletal heatmaps. Given an image in Fig. 1(a), skeletal attention module produces skeletal heatmaps as shown in Fig. 1(b). Then, by combining the skeletal heatmaps, three body attention maps for upper-body, lower-body, and full-body in Fig. 1(c), (d), and (e), respectively, are generated. Finally, we use the skeletal heatmaps and the three body attention maps to yield the output heatmaps for keypoints, as shown in Fig. 1(f). Experimental results demonstrate that the proposed algorithm outperforms conventional algorithms on two benchmark datasets [19], [20].

This work has the following main contributions:

- We develop the novel skeletal attention module that produces skeletal heatmaps and body attention maps to represent adjacency information between keypoints.
- We further improve the estimation performance by estimating keypoint locations from heatmaps based on the weighting of multiple candidates.
- The proposed algorithm outperforms the baseline algorithm in the COCO val2017 [19] and MPII validation [20] benchmark datasets.

The rest of this paper is organized as follows. Section 2 describes the proposed algorithm. Section 3 discusses and evaluates the performance of the proposed algorithm comparatively with conventional algorithms. Finally, Section 4 concludes this work.

II. Proposed Algorithm

We aim to train the human pose estimator f. Given an input RGB image I with size $h \times w \times 3$, f outputs a set of keypoint heatmaps H.

$$f : I \longrightarrow H, \quad H = \{H_i\} \tag{1}$$

where H_i denotes the heatmap for the ith keypoint. To generate the ground-truth H_i, we follow the method of [12], [13]. We first produce the binary image, which is filled in with all 0s except for a single 1 on the keypoint coordinate. H_i is obtained by applying the Gaussian filter to this image.

To improve the training of the human pose estimator, we design the skeletal attention module that produces two kinds of maps: skeletal heatmaps and body attention maps. Fig. 2 shows an overview of the proposed human pose estimator and examples of estimated maps. First, to extract the features from the input image, we adopt HRNet [14] as the backbone network. The skeletal attention module is attached after the third stage of HRNet. Except for the proposed skeletal attention module, the rest of the network is the same as [14]. The skeletal attention module estimates skeletal heatmaps H^{s} using the features from the backbone network, as shown in Fig. 2(a) and (b). Body attention maps are produced by combining H^{s}. We construct three body attention maps: upper body, lower-body, and full-body attention maps H^{u}, H^{l}, and H^{f}, which

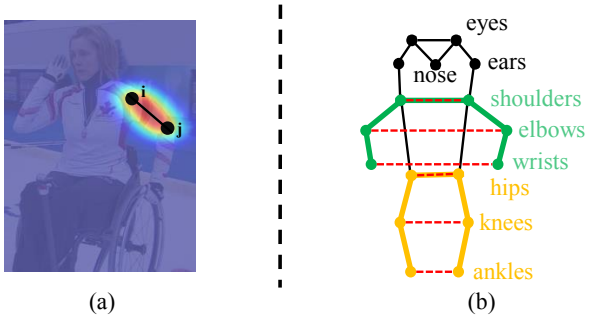


Fig. 3. An example of a ground-truth skeletal heatmap connecting keypoint i and j is shown in (a). In (b), green and yellow lines correspond to physical skeletal heatmaps, while dashed red lines to virtual skeletal heatmaps, connecting left-right symmetric keypoints. Also, note that the green lines and the top three dashed lines belong to the upper body, while the others to the lower body.

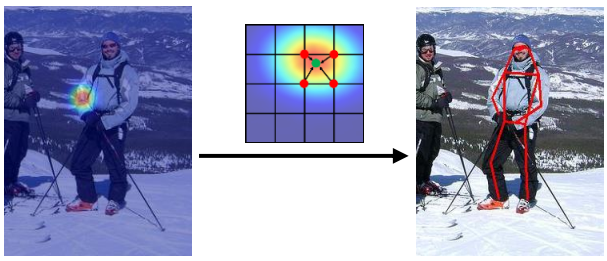


Fig. 4. Weighted inference of a keypoint for $N = 4$.

are shown in Fig. 2(c), (d), and (e), respectively. The skeletal heatmaps and the body attention maps are combined with backbone features through elementwise summation operation. Finally, the last stage of the network predicts a set $\{H_i\}$ of heatmaps for keypoints from the combined features. For more precise inference of the final keypoint coordinates, we perform the heatmap weighting.

A. Skeletal Heatmaps

Each skeletal heatmap H^{s}_{ij} represents the locational information of the line connecting two keypoints i and j. To generate H^{s}_{ij}, we produce a binary image where the pixels corresponding the line connecting i and j as 1, and the others are 0. Then, by applying a gaussian filter with standard deviation 1 to this binary image and clipping the maximum value to 1, we obtain H^{s}_{ij}. Fig. 3(a) shows an example of skeletal heatmap.

Fig. 3(b) shows the links for the skeletal heatmaps. First, green and yellow links correspond to physical bones of upper-body and lower-body, respectively. Ten physical skeletal heatmaps are defined in total. Second, using symmetric relation in an image similar to [21], we generate virtual links by connecting left-right symmetric keypoints (*e.g.* left and right wrists) which are denoted by red dashed lines. Six such virtual skeletal heatmaps are adopted in total. The virtual heatmaps help to distinguish the left and right sides more precisely. Note that we define the skeletal heatmaps for the upper and lower body parts only, not for the head.

B. Body Attention Maps

Body attention maps are generated by combining multiple skeletal heatmaps. The upper-body attention map H^{u} and the lower-body attention map H^{l} are given by

$$H^{\mathrm{u}} = \sum_{(i,j)\in U} H^{\mathrm{s}}_{ij}, \quad H^{\mathrm{l}} = \sum_{(i,j)\in L} H^{\mathrm{s}}_{ij} \tag{2}$$

where U and L denote the set of the keypoint pairs belong to the upper-body and the lower-body. We then define the full-body attention map H^{f} as the sum of H^{u} and H^{l}.

$$H^{\mathrm{f}} = H^{\mathrm{u}} + H^{\mathrm{l}}. \tag{3}$$

C. Skeletal Attention Module

The skeletal attention module is shown in Fig. 2. It receives the backbone feature from the third stage of HRNet [14] as input and outputs the skeletal heatmaps H^{s}. Then, H^{u}, H^{l}, and H^{f} are generated as stated above and added elementwise to the backbone feature of the third stage. Specifically, H^{s} is added to the highest resolution feature, H^{u} and H^{l} to the second-highest one, and H^{f} to the third-highest one. The attention features are replicated to match the channel size between the attention features and the backbone features. Let T denote the number of channels on the backbone feature tensor. Then, the skeletal heatmaps are replicated $T/16$ times, the half-body attention maps are replicated $T/2$ times, and the full-body attention maps are replicated T times. To match the resolution between tensors, the width and height of H^{u} and H^{l} are halved, and H^{f} is downsized with a factor of $\frac{1}{4}$.

D. Weighted Inference of Keypoints

In the conventional methods [12], [13], the location of the keypoint is determined as the coordinates of the maximum value in the heatmap. However, an estimator does not always generate a heatmap with a single clear peak. For example, if there is occlusion in a scene, the location of a heatmap peak may not match a keypoint location. Also, if a target person is too small, the pixel level prediction can be inaccurate. To alleviate these problems, we use multiple coordinates with high values on the heatmap to determine the final keypoint coordinates. From an estimated heatmap H_i, The predicted coordinate vector $\hat{\mathbf{c}}_i$ is calculated as follows.

$$\hat{\mathbf{c}}_i = \sum_{n=1}^{N} w_n \times \mathbf{c}_{in} \tag{4}$$

where $\mathbf{c}_{in}$ denotes the coordinate vector for the nth largest value in H_i, and w_n is the weight for $\mathbf{c}_{in}$. w_n is determined by its rank n as follows.

$$w_n = \frac{1}{n} \left(\sum_{h=1}^{N} \frac{1}{h} \right)^{-1}, \tag{5}$$

Input size	Method	#Params	GFLOPs	mAP	AP (.5)	AP (.75)	AP (M)	AP (L)	mAR
256×192	Baseline(HRNet-w32)	28.5M	7.1	76.5	93.5	83.7	73.9	80.8	79.3
	Proposed(HRNet-w32)	29.3M	7.2	76.8	92.5	83.8	73.9	81.0	79.4
	Baseline(HRNet-w48)	63.6M	14.6	77.1	**93.6**	84.7	74.1	81.9	79.9
	Proposed(HRNet-w48)	65.3M	14.8	**77.9**	**93.6**	**84.8**	**75.1**	**82.0**	**80.6**
384×288	Baseline(HRNet-w32)	28.5M	16.0	77.7	93.6	84.7	74.8	82.5	80.4
	Proposed(HRNet-w32)	29.3M	16.2	78.2	93.5	84.7	75.0	82.9	80.7
	Baseline(HRNet-w48)	63.6M	32.9	78.1	**93.6**	**84.9**	**75.3**	83.1	80.9
	Proposed(HRNet-w48)	65.3M	33.3	**78.4**	**93.6**	84.6	75.1	**83.3**	**81.0**

TABLE I: Performance comparison in terms of AP scores on the COCO val2017 dataset. The highest score for each input size setting is boldfaced.

Method	PCKh@0.5	PCKh@0.1
Baseline(HRNet-w32)	90.3	37.7
Proposed(HRNet-w32)	**90.6**	39.6
Baseline(HRNet-w48)	90.5	39.8
Proposed(HRNet-w48)	**90.6**	**39.9**

TABLE II: Performance comparison in terms of PCKh on the MPII validation dataset. The best result is boldfaced.

Method	PCKh@0.5	PCKh@0.1
HRNet-w32	90.3	37.7
+ skeletal attention module	90.5	38.1
+ weighted inference	**90.6**	**39.6**

TABLE III: Ablation study on the MPII validation dataset. The best result is boldfaced.

where $\sum_{n=1}^{N} w_n = 1$. We experimentally set $N = 20$ to include the top 20 largest points. Fig. 4 illustrates this weighted inference for $N = 4$.

III. Experimental Results

A. Datasets

COCO [19] is a common dataset for various tasks, including object detection, segmentation, and keypoint detection. For keypoint detection, it provides over 200K images and 250K person instances. The images are divided into three parts; train2017, val2017, and test-dev2017, which include about 57K, 5K, and 20K images, respectively. We train the proposed algorithm on the train2017 split and assess it on the val2017 split. The annotation labels in the COCO dataset consist of up to 17 keypoints per person. To compare the performance only for pose estimation, we compare the results using the ground-truth bounding boxes of person instances.

The MPII dataset [20] is also commonly used for human pose estimation, which consists of 25K images with 40K subjects. Unlike COCO, the annotations in MPII consist of up to 16 keypoints per person. Following the setting in [14], we use 22K images for training and use 3K images for validation.

We use input sizes 256×192 and 384×288 for the COCO dataset, and 256×256 for the MPII dataset.

B. Evaluation Metrics

For the COCO dataset, we use two types of metrics: mean average precision (mAP) and mean average recall (mAR), based on the object keypoint similarity (OKS). Note that, for the object detection task [22], [23], mAP and mAR are computed by varying the intersection over union (IOU) ratio. For the keypoint detection task, OKS plays the same role as IOU and is defined as

$$\text{OKS} = \frac{\sum_i \exp\left(\frac{-d_i^2}{2s^2k_i^2}\right)\mathbb{I}(v_i > 0)}{\sum_i \mathbb{I}(v_i > 0)} \tag{6}$$

where $\mathbb{I}$ is the indicator function, and d_i is the Euclidean distance between keypoint i and its ground-truth. Also, v_i is the visibility of keypoint i, s is the scale of the object, and k_i is a per-keypoint constant pre-defined in [19]. The precision and recall are defined as

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{7}$$

where TP, FP, and FN denote the numbers of true positives, false positives, and false negatives, respectively. We measure AP scores for 10 OKS values from 0.5 to 0.95 with an interval of 0.05 and set mAP as the mean of the 10 AP scores. AP (.5) and AP (.75) denote the AP scores for OKS values from 0.5 and 0.75. AP (M) and AP (L) are the mAP scores for medium and large objects.

For the MPII dataset, we use PCKh@0.5 and @0.1 [20] for evaluation metric. They measure whether a predicted keypoint is located within a circle around the ground-truth keypoint. The radius of the circle is determined by multiplying the diagonal length of the bounding box of the annotated person's head with a factor. Here @0.5 and @0.1 indicate the factors.

C. Training Details

We compare the proposed algorithm with the baseline network HRNet [14]. We add the proposed skeletal attention module to the baseline network and train it in the same way as [14]. Specifically, we use the Adam optimizer [24] with an initial learning rate of 10^{-3} and a weight decay 10^{-4} for both

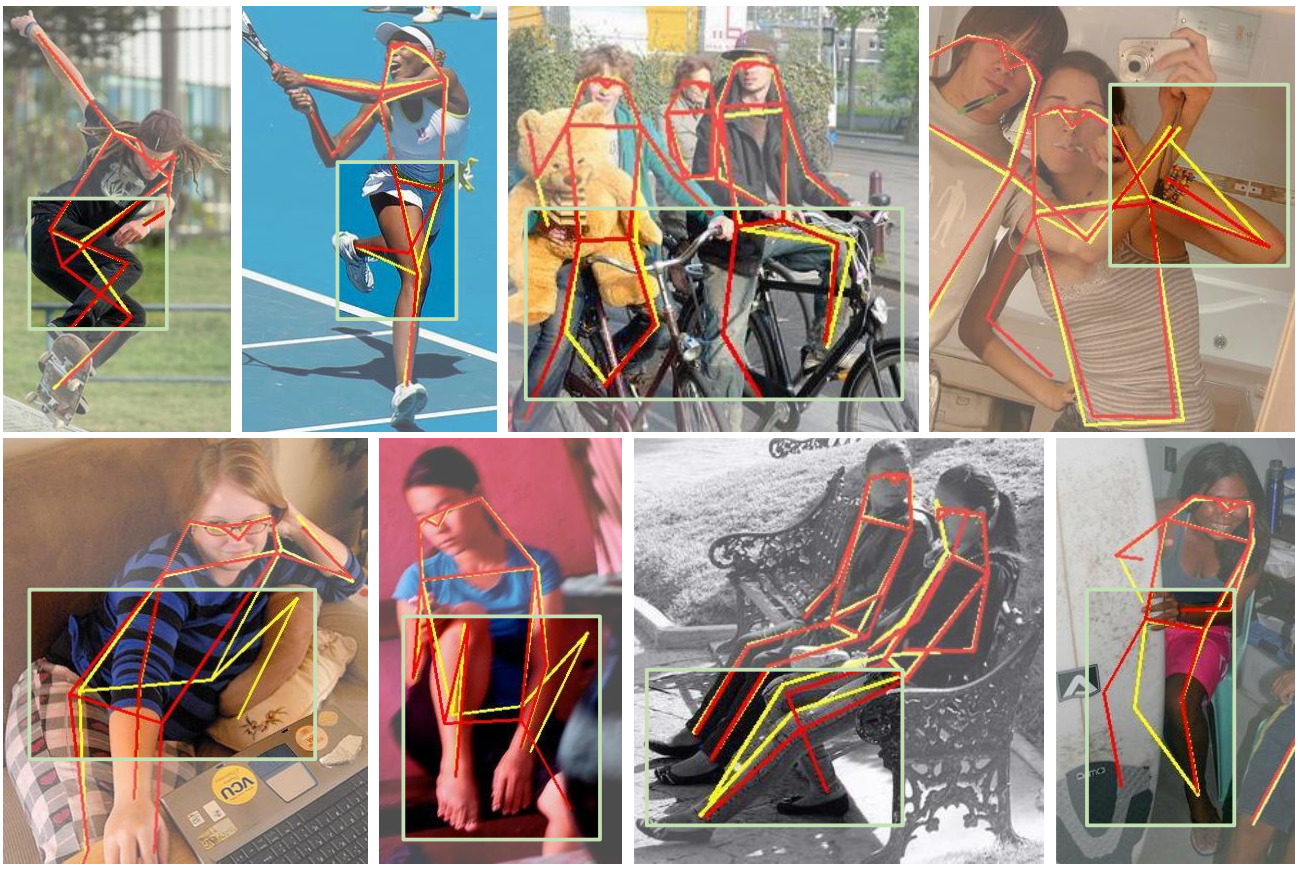


Fig. 5. Qualitative comparison of keypoint estimation results of HRNet-w32 (yellow lines) and the proposed algorithm (red lines) on the COCO val2017 dataset.

networks. We drop the learning rate to 10^{-4} and 10^{-5} at the 170th and 200th epochs, respectively.

We use multiple loss functions to boost the network training. It is demonstrated in prior work [25] that exploiting multiple loss functions improves the training results. The keypoint heatmap loss l_{k} computes the squared errors between ground-truth keypoint heatmaps and predicted ones. Similarly, the skeletal heatmap loss l_{s} is computed between ground-truth skeletal heatmaps and their predicted counterparts. The two loss functions are given by

$$l_{\mathrm{k}} = \sum_{i} \|\hat{H}_i - H_i\|^2, \quad l_{\mathrm{s}} = \sum_{(i,j)\in S} \|\hat{H}^{\mathrm{s}}_{ij} - H^{\mathrm{s}}_{ij}\|^2 \tag{8}$$

where i and j are keypoints. Also, S is a set of keypoint index pairs, and the elements of S correspond to the colored lines in Fig. 3(b). $\hat{H}_i$ and $\hat{H}^{\mathrm{s}}_{ij}$ denote the estimated heatmaps for the ground-truth $\hat{H}_i$ and $\hat{H}^{\mathrm{s}}_{ij}$, respectively. We compute l_{k} only for visible keypoints in input images, and compute l_{s} only if both keypoints i and j are visible.

The total loss l for the network is defined as a weighted sum, given by

$$l = l_{\mathrm{k}} + \lambda l_{\mathrm{s}} \tag{9}$$

where λ controls the importance between the two losses. It is set to 0.2 in this work. We use flipping, rotation, and scaling for data augmentation, as done in [14].

D. Comparison Results

We compare the proposed algorithm with the state-of-the-art algorithm, HRNet [14]. Whereas many conventional algorithms, such as [26], use typical encoder-decoder networks, HRNet attempts to preserve high-resolution features and obtain low-resolution features using additional branches.

Table I compares the results on the COCO val2017 dataset, and Table II compares the results on the MPII validation dataset. On both datasets, the proposed algorithm outperforms HRNet, by adding a moderate number of parameters. Table III shows how the proposed skeletal attention and the weighted inference improve the performance. Note that the performance improvement on PCKh@0.1 is large, which indicates that the proposed algorithm makes more precise predictions.

Fig. 5 compares qualitative results. It can be seen clearly in the green box of each image that the proposed algorithm locates keypoints more precisely than HRNet. More specifically, the proposed algorithm provides better predictions on occluded scenes and outmost keypoints, such as the wrist and

ankle. The proposed algorithm distinguishes the left and right sides accurately and represents the skeletal structure of humans faithfully.

IV. Conclusions

We proposed the skeletal attention module to improve the performance of human pose estimation. We designed the pose estimation network by adding the skeletal attention module to a backbone network. The skeletal attention module was trained to generate skeletal heatmaps. Also, three body attention maps were generated, by combining multiple skeletal heatmaps. The generated skeletal heatmaps and body attention maps were added with the features of the backbone network and then used to estimate the heatmap of each keypoint. Finally, we used heatmap weighting to predict the locations of keypoints more precisely. Experimental results demonstrated that the proposed algorithm outperforms the conventional algorithm.

V. Acknowledgment

This work was supported by 'The Cross-Ministry Giga KOREA Project' grant funded by the Korea government (MSIT) (No. GK20P0200, Development of 4D reconstruction and dynamic deformable action model based hyperrealistic service technology), and by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. NRF-2018R1A2B3003896).

References

[1] G. Chéron, I. Laptev, and C. Schmid, "P-CNN: Pose-based CNN features for action recognition," in *ICCV*, 2015.
[2] Y. J. Koh, Y.-Y. Lee, and C.-S. Kim, "Sequential clique optimization for video object segmentation," in *ECCV*, 2018.
[3] H.-U. Kim and C.-S. Kim, "CDT: Cooperative detection and tracking for tracing multiple objects in video sequences," in *ECCV*, 2016.
[4] C. Zimmermann, T. Welschehold, C. Dornhege, W. Burgard, and T. Brox, "3D human pose estimation in RGBD images for robotic task learning," in *ICRA*, 2018.
[5] P. F. Felzenszwalb and D. P. Huttenlocher, "Pictorial structures for object recognition," *Int. J. Comp. Vis.*, vol. 61, no. 1, pp. 55–79, 2005.
[6] M. Andriluka, S. Roth, and B. Schiele, "Pictorial structures revisited: People detection and articulated pose estimation," in *CVPR*, 2009.
[7] X. Ren, A. C. Berg, and J. Malik, "Recovering human body configurations using pairwise constraints between parts," in *ICCV*, 2005.
[8] T.-P. Tian and S. Sclaroff, "Fast globally optimal 2D human detection with loopy graph models," in *CVPR*, 2010.
[9] X. Lan and D. P. Huttenlocher, "Beyond trees: Common-factor models for 2D human pose recovery," in *ICCV*, 2005.
[10] A. Toshev and C. Szegedy, "DeepPose: Human pose estimation via deep neural networks," in *CVPR*, 2014.
[11] J. Carreira, P. Agrawal, K. Fragkiadaki, and J. Malik, "Human pose estimation with iterative error feedback," in *CVPR*, 2016.
[12] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *ECCV*, 2016.
[13] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," in *CVPR*, 2016.
[14] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," in *CVPR*, 2019.
[15] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multiperson 2D pose estimation using part affinity fields," in *CVPR*, 2017.
[16] 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 *ECCV*, 2018.
[17] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei, "Fully convolutional instance-aware semantic segmentation," in *CVPR*, 2017.
[18] M. Heo, J. Lee, K.-R. Kim, H.-U. Kim, and C.-S. Kim, "Monocular depth estimation using whole strip masking and reliability-based refinement," in *ECCV*, 2018.
[19] 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 *ECCV*, 2014.
[20] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, "2D human pose estimation: New benchmark and state of the art analysis," in *CVPR*, 2014.
[21] D. K. Jin, J.-T. Lee, and C.-S. Kim, "Semantic line detection using mirror attention and comparative ranking and matching," in *ECCV*, 2020.
[22] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards realtime object detection with region proposal networks," in *NIPS*, 2015.
[23] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *ICCV*, 2017.
[24] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015.
[25] J.-H. Lee and C.-S. Kim, "Multi-loss rebalancing algorithm for monocular depth estimation," in *ECCV*, 2020.
[26] B. Xiao, H. Wu, and Y. Wei, "Simple baselines for human pose estimation and tracking," in *ECCV*, 2018.