Contents lists available at ScienceDirect

Signal Processing: Image Communication

journal homepage: www.elsevier.com/locate/image

Feedback-based object detection for multi-person pose estimation

Jaeseo Park, Junho Heo, Suk-Ju Kang [*]

Department of Electronic Engineering, Sogang University, Seoul 04107, Republic of Korea

ARTICLE INFO

Keywords:
Pose estimation
Object detection
Multi-person
Real-time system

ABSTRACT

In this paper, a novel method is proposed for increasing the performance through coupling of top-down models adjusting the object detector based on a new loss function. Generally, object detectors and keypoint estimators are sequentially used in real-time multi-person pose estimations; however, these two models are separately trained. Therefore, the results of the object detector are not optimized for the keypoint estimator. To solve this problem, we analyze the relationship between the two models and propose a feedback-based loss optimization in the object detector, based on the estimation results of the keypoint estimator. In addition, the resulting bounding box of the object detector is readjusted to improve the accuracy of the keypoint estimation model. The experimental results demonstrate that the proposed approach can perform real-time operations with a high frame rate similar to that of the baseline model. Moreover, it achieved an accuracy of 74.2 average precision (AP), which is higher than the state-of-the-arts model including the human detector used in the experiment.

1. Introduction

Multi-person pose estimation (MPPE) is a challenging problem in the field of computer vision owing to various gestures and unpredictable interactions of humans. The key technique in MPPE is to estimate the keypoints for all humans in an image. This approach has been utilized in many visual applications such as human behavior recognition and human–computer interaction. Classical approaches for MPPE formulate the problem of keypoint estimation mainly based on a tree structure and graphical model [1–4]. Something of the concepts applied in these classical approaches have been used in recent studies. The performance of MPPE has been significantly improved through the use of convolutional neural networks (CNNs) [5–7]. As an example, Cao et al. [7] used convolutional pose machines [8] to find keypoint joints in an image. In general, MPPE using a CNN can be broadly divided into two approaches: top-down [9–14] and bottom-up [15,16].

Top-down approaches estimate the keypoints in a two-stage pipeline. In [17], all person instances in an image are detected and fed to a single person pose estimation (SPPE) model individually. Specifically, the first stage predicts the location and scale of the boxes that are likely to contain people. In the second step, keypoints are predicted for each of the detected boxes. In addition, modules for upgrading the model have been previously studied. DarkPose [18] adopted plug-in modules that encode processes by generating accurate heatmap distributions for unbiased model training. Furthermore, this technique was applied to HRNet [19,20] to verify its performance. In recent studies, top-down approaches have been shown to exhibit a higher accuracy than bottom-up approaches. However, top-down models are relatively slow in terms of their operation time [17,21,22]. Because a keypoint estimation is applied for each detected object, the processing time is proportional to the number of people in an input image.

A bottom-up approach involves detecting partial regions of the human body and connecting those regions to form a human instance. Because this approach uses partial region detectors, various relationships that can exist between the detected partial regions of the human body must be considered. OpenPose [15] maps the relationship between keypoints to a part affinity field (PAF) and assembles the detected keypoints into human poses. PAFs are 2D vector fields that encode the location and orientation of limbs over the image domain. A bottom-up approach remains fast even when the number of people in the input image increases. However, its accuracy is low compared to that of a top-down approach, and improving its performance is difficult [15,16] owing to the high complexity of mapping joints to the corresponding individuals.

Recently, a lot of research has been done to improve the SPPE model using existing models such as Resnet [6] and HRNet by applying a new method. DarkPose [18] discussed above is one such study. Also, among recent studies, EvoPose2D [23] presents the weight transfer technique to relax the function-preserving mutations, and it can accelerate neuro-evolution in a flexible manner. In addition, AID [24] propose customized training schedules, which are designed by analyzing the pattern of loss and performance in training process from the perspective of information supplying. Likewise, also in our study, we present an approach to adapt and improve existing models.

[*] Corresponding author.
E-mail addresses: jspark3@sogang.ac.kr (J. Park), jayceheo92@gmail.com (J. Heo), sjkang@sogang.ac.kr (S.-J. Kang).

https://doi.org/10.1016/j.image.2021.116508
Received 23 January 2021; Received in revised form 4 September 2021; Accepted 16 September 2021
Available online 23 September 2021
0923-5965/© 2021 Published by Elsevier B.V.

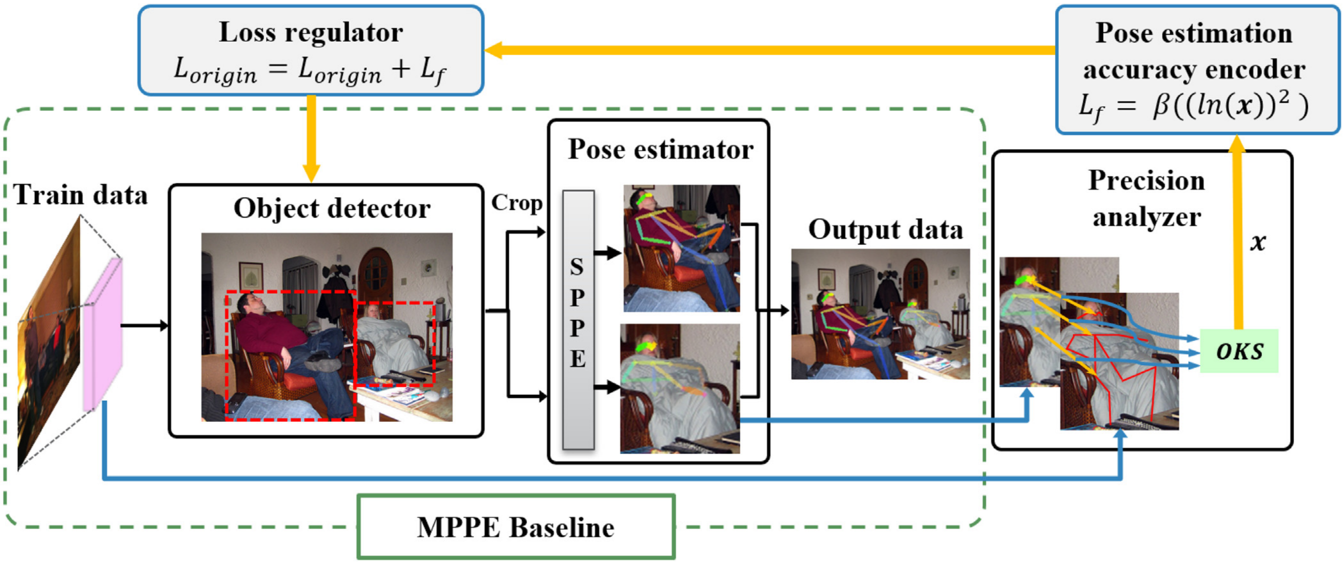

Fig. 1. Overall architecture of the proposed feedback-based loss optimization system. The flowchart inside the green dotted line indicates the system when used as a baseline. Further learning is carried out through items of the blue arrow and yellow arrow. The resulting inference operates in the same way as the baseline system.

MPPE has recently been used for real-time applications such as vehicle cameras and closed-circuit television (CCTV). Therefore, its performance must be considered in terms of both speed and accuracy. Recently, there have been attempts to replace slow object detectors that conduct a thorough search [18] with a faster model that detects all objects within the image grid simultaneously [25–28]. Compared to slow models, this model is up to 30 times faster but offers approximately 1.5 times lower accuracy [29]. Because the results of the SPPE depend entirely on the quality and resolution of the input image, errors in the object detector significantly impair the performance of the overall system. However, most MPPE studies simply link SPPE to existing object detectors. Moreover, the two models are independently trained and evaluated, thereby increasing the possibility of SPPE errors. In this paper, we propose a solution applied to object detection to overcome this limitation and to inherit the advantages of the fast top-down model approach.

In this research, we focus on the relevance and optimization of the two models. Based on the results of the keypoint estimator, we apply feedback-based loss optimization to the previous models, as shown in Fig. 1. We also propose a method of bounding box expansion to facilitate keypoint estimation, as shown in Fig. 2. The main contributions of this paper are as follows:

· We propose a novel loss function in which the pose estimation accuracy is fed back to the bounding box detector.

• We present a novel algorithm that appropriately expands the detected bounding boxes, taking into account the detected positions and number of individuals. This algorithm mitigates the propagation of an incorrect output of the bounding box detector to the SPPE.

2. Proposed methods

Figs. 1 and 2 show the overall architecture of the proposed system. The black arrows show the application of the proposed loss function to the object detector. In addition, the blue and yellow arrows show the operation of the proposed detection expander. The methods used in our system are described in detail below.

2.1. Feedback-based loss optimization

The accuracy is evaluated using a distance-based metric called object keypoint similarity (OKS), which applies measurements differently based on the keypoint type. In MS-COCO [30], the distance between the prediction and ground-truth is calculated using the OKS with a type variant parameter, and these parameters are stored for each image. In this study, the feedback-based loss (L_f) of the object detector is calculated using the parameters and the final output of the entire system. Considering the penalty, L_f is formulated as follows:

$$L_f = \beta\left((\ln(x))^2\right), x \in \mathrm{D}, \tag{1}$$

where the prediction precision, D, is read back from the object detector trainer. The score of each image is calculated as L_f. In this manner, the merging between the models is not performed directly. Here, L_f is added to the existing mean squared error (MSE), which is the original loss. The keypoint prediction missing rate is a reference value for determining the penalty of the loss function. The parameter β"-, which is used to adjust the strength of this penalty, is determined by a design choice with the experiments. Our final loss function applying (1) is as follows.

$$Total\ \ Loss = L_{origin} + \beta\left\{\ln\sum_{j=0}^{D}\frac{\sum_{ij}\exp\left(-d_{ij}^2/2s^2k_{ij}^2\right)\delta\left(v_{ij}>0\right)}{\sum_{ij}\delta\left(v_{ij}>0\right)}\right\}^2, \tag{2}$$

where d_{ij} is the euclidean distance between the detected keypoint and the corresponding ground truth, v_{ij} is the visibility flag of the ground truth, s is the object scale, and k_{ij} is a per-keypoint constant that controls fall-off. The scale and keypoint constant are needed to equalize the importance of each keypoint because the neck location is more precise than hip location. We used the fast and popular object detector [25] for our models. (3) is a conceptual equation of the loss function finally used in our model. The pose estimation feedback loss is what we added.

$$\begin{aligned} Total\ \ Loss = {} & Regression\,(bounding\ \ box)\,loss \\ & + Confidence\ \ loss + Classification\ \ loss \\ & + Pose\ \ estimation\ \ feedback\ \ loss\,(Ours)\,. \end{aligned} \tag{3}$$

Finally, this modified loss function leads to the learning of the object detector and prevents the object detector from feeding a high predicted value when the output of the pose estimator is low. As demonstrated in Table 2, this change results in a performance improvement. Interestingly, to the best of our knowledge, there have been no previous studies on the impact of this change on human posture estimation performance.

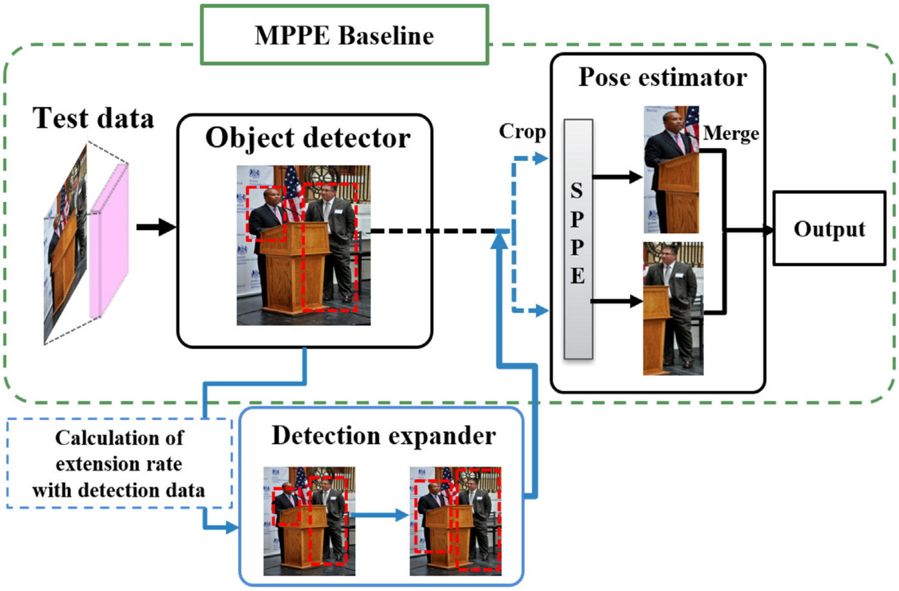

Fig. 2. Overall architecture of bounding box optimization through the bounding box expansion system. The blue arrows indicate the flow of the object detection expander, which calculates the expansion rate based on the object detection data.

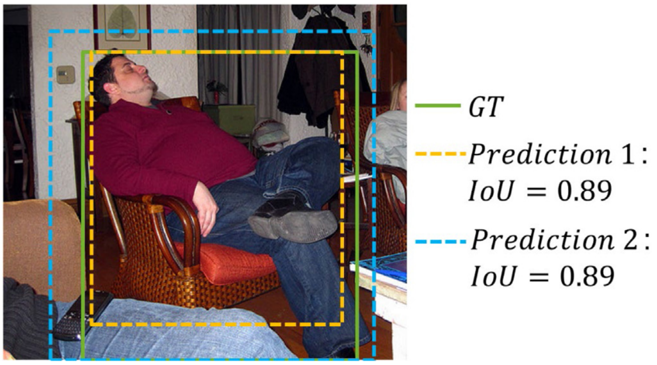

Fig. 3. Different bounding boxes with the same intersection over union (IoU) value. In the typical IoU, loss of image features is not considered.

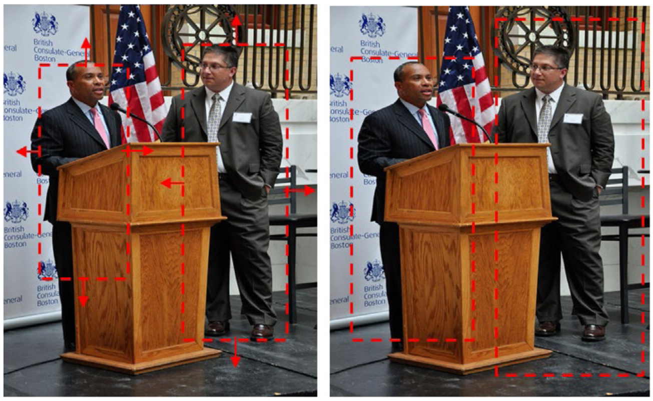

Fig. 4. Target and output images of the proposed system. As shown in the image on the left, the bounding box of the original object detector is extremely tight and part of the body is outside the bounding box. To accurately estimate the pose, the object detection results must be appropriate and include all keypoint features, as shown in the image on the right.

Table 1
Ablation studies on COCO2017.

Parameters	Accuracy (AP)	Parameters		Accuracy (AP)
Feedback-based loss optimization		Bounding box expansion		
$\beta = 10$	74.8	$\gamma = 5$	$\mu = 10$	75.3
$\beta = 5$	75.0	$\gamma = 4$	$\mu = 6$	75.6
$\beta = 2$	75.6	$\gamma = 2$	$\mu = 4$	75.4
$\beta = 1$	75.0	$\gamma = 2$	$\mu = 3$	75.0

2.2. Detection expander

In the top-down approach, there is a problem in that the inappropriate results of the object detector propagate to the SPPE. The existing approach does not provide the keypoint estimator with a perfectly appropriate object bounding box. To overcome this problem, in this study, an object bounding box extension is devised that reduces the omission of the area containing the keypoint information in the image. As shown in Fig. 3, there are cases in which an important area can be excluded even with the same intersection over union (IoU). In this study, we extend the bounding box to reduce the loss of features for the keypoint estimation step. In addition, if the number of people detected increases, there is a higher probability of including another person in the expanded bounding box. Therefore, the number of people in the image is considered as a variable parameter (see Fig. 4).

The variable specifying the range to be expanded based on the number of detected people is labeled α, and the function of α is described as follows:

$$\alpha = \log(n(p))\gamma + \mu, \;\; p \in P, \tag{4}$$

where P is the set of bounding boxes output from the object detector, and p is the set of bounding boxes in one of the input images. In addition, α indicates how far the bounding box will expand, and it is a monotonic function that must decrease with increase in the value of p. In addition, γ and μ show how rapidly α is penalized. These hyper-parameters are experimentally determined. The calculated expansion factor α is multiplied by all values of p.

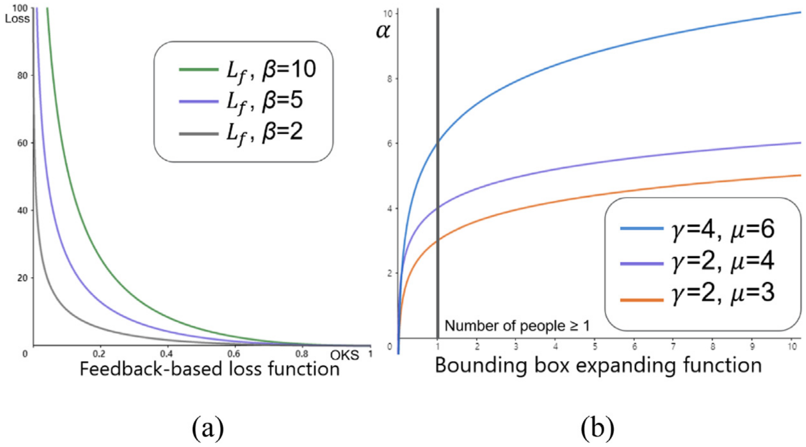

Fig. 5. Functional graphs when changing the hyperparameter values. The sensitivity of the function is controlled by adjusting the hyperparameters. This figure shows the change in the loss value according to the change in the hyperparameter. (a) indicates that as the parameter value increases, the amount of loss reduction from oks rapidly increases, and (b) illustrates the box expansion rate, which needs to be adjusted according to the number of people in the image.

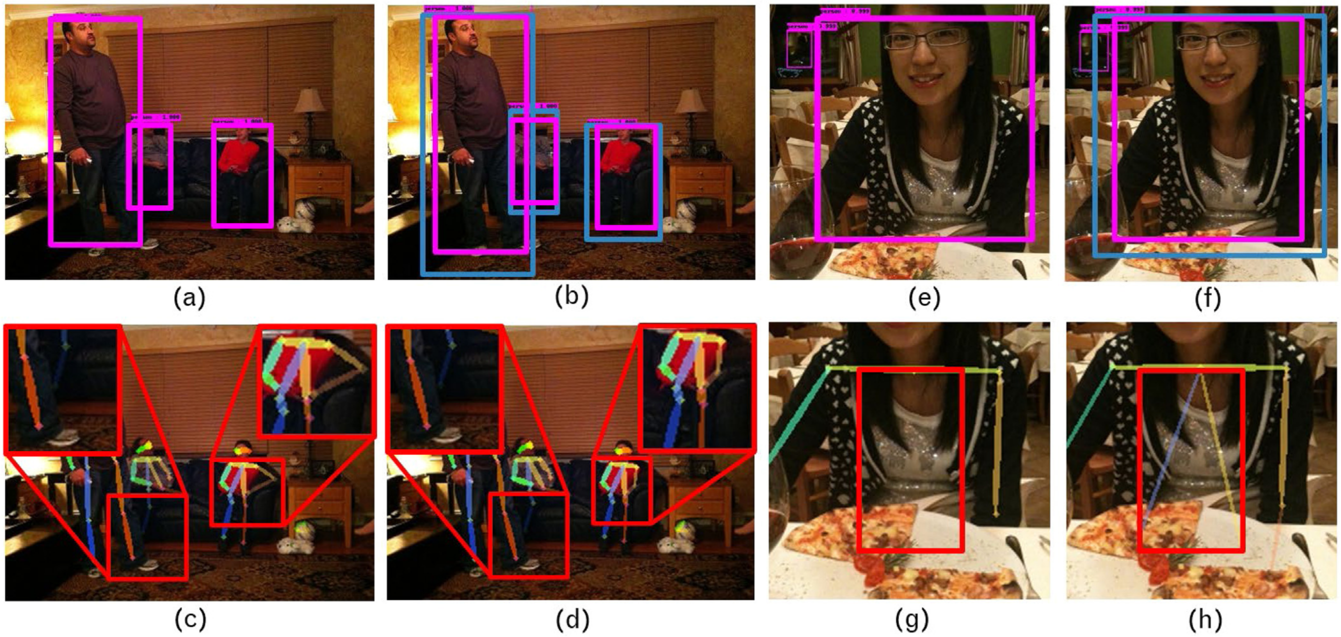

Fig. 6. Object detector (upper) and human pose estimator (bottom) outputs for each image set (the left is from an original image and the right is the image resulting from our method): (a) and (e) are the results of the original object detector, (b) and (f) are the results of applying the proposed bounding box expansion method to the object detector, and (c) and (g) are the results of the poses estimated using the original object detector. Red boxes in the images mark important changes.

3. Network implementation

3.1. Baseline network

The networks used as the baseline for this study were the human detector and SPPE developed using the PyTorch library. The details of each model and the performance of the baseline network are as follows. Two models were tested in this study. The first human pose estimator was HRNet-W48 [19,20], the input of which had a 384 × 288 pixel resolution. This network has an average precision (AP) of 73.9 when using our baseline human object detector with the COCO val2017 dataset. The second human pose estimator was Table 2 based on the original regional multi-person pose estimation (Rmpe) [26]. This model was a little less accurate, but faster than the first model. The final MPPE network of this model had an AP of 73.0 for the COCO val2017 dataset. Among the models shown in Tables II, Rmpe, the baseline MPPE, and our models used the same human object detector, i.e., Yolov3 [25]. Pre-trained weights were used to predict the images. This human object detector had a human AP of 55.5 for the COCO val2017 dataset. Table 2 shows the inference time and accuracy when using an image adjusted to 384 × 288 and 320 × 320 pixel resolutions as an input.

3.2. Implementation details

3.2.1. HRNet based model

The performance of the baseline MPPE network was 73.9 AP when an input is 320 × 228 pixel resolution.

In addition, the frame rate was 11.2 fps when the average number of people present in the images was four. This demonstrates the execution speed in a low-efficiency Python environment when using a single Titan XP1 GPU. For HRNet [19,20] and DARK (+HRNet-W48) [36], we applied the pre-trained models as baseline models for our networks. The total number of epochs in HRNet was 140. We used two different input sizes (384 × 288 and 320 × 320 pixel resolutions) in our experiment.

Table 2
Validation results on COCO2017 keypoint validation dataset.

Network	Input size	#params	Accuracy					
			AP	AP 0.50	AP 0.75	AP 0.95	AP^M	AP^L
Bottom-up methods								
OpenPose(max) [15]	–	–	65.3	85.2	71.4	–	57.1	68.2
Pifpaf [31]	321 × 321	–	66.7	–	–	–	62.4	62.4
SPM [32]	384 × 384	–	66.9	88.5	72.9	–	62.6	–
Higher HRNet -W48 [33]	640 × 640	63.8M	68.4	88.2	75.1	–	64.4	74.2
Top-down methods								
Mask RCNN [17]	–	–	63.1	87.3	68.7	57.8	71.4	71.4
Simple Baseline [34]	384 × 288	68.6M	73.7	91.9	81.1	–	70.3	80.0
Rmpe[a] [26]	320 × 256	28.1M	73.0	87.5	79.4	28.8	68.2	78.7
EvoPose2D-M [23]	384 × 288	7.34M	75.1	90.2	81.9	-	71.5	81.7
HRNet -W48[a] [20, 32]	384 × 288	63.6M	73.9	89.2	80.6	–	70.1	80.0
DARK (+ HRNet -W48)[a] [35]	384 × 288	63.6M	74.1	89.3	80.9	–	70.3	80.4
Ours methods								
Feedback-based loss optimization (rmpe + ours)[a]	320 × 320	28.1M	73.1	86.0	80.8	27.9	–	–
Bounding box expansion (rmpe + ours)[a]	320 × 320	28.1M	73.6	88.1	80.0	28.3	–	–
Rmpe + final model[a]	320 × 320	28.1M	73.5	87.5	80.8	27.9	68.4	79.6
Feedback-based loss optimization (hrnet -w48 + ours)[a]	320 × 228	63.6M	73.7	89.0	80.7	29.7	70.0	80.7
Bounding box expansion (hrnet-w48 + ours)[a]	320 × 228	63.6M	74.0	89.8	80.6	29.4	70.2	80.9
Hrnet-w48 + final model[a]	320 × 228	63.6M	74.1	89.6	80.5	29.5	70.3	81.1
Feedback-based loss optimization (dark hrnet -w48 + ours)[a]	320 × 228	63.6M	73.8	89.1	81.0	29.8	69.8	80.8
Bounding box expansion (dark hrnet-w48 + ours)[a]	320 × 228	63.6M	74.2	89.5	80.8	29.4	70.4	80.7
Dark hrnet-w48 + final model[a]	320 × 228	63.6M	74.0	89.3	80.9	29.8	70.3	80.7

[a] The figures may differ from the figures in the content of the paper. We experimented with resizing the image in our environment.

3.2.2. Rmpe based model

The performance of the baseline MPPE network was 73.0 when the input had 320 × 320 pixel resolution. In addition, the frame rate was 23 fps when the average number of people present in the images was four. This demonstrates the execution speed in a low-efficiency Python environment when using a single Titan XP1 GPU. For the Rmpe [26] model with Yolov3 [25], we followed the same learning schedule and epochs as in our original study.

4. Experimental results

Our system followed a top-down approach in applying MPPE. The proposed feedback-based loss optimization and detection expander were applied to the MPPE, and each hyperparameter was changed. This section describes the results of these experiments and the details of the proposed method. In the table, items with a high accuracy are indicated in bold.

4.1. Datasets and settings

Our model was trained on the COCO2017 [30] training dataset and evaluated on the COCO2017 keypoint validation dataset with 5000 images. The COCO keypoint dataset consisted of imagery data with various human poses, unconstrained environments, different body scales, and occlusion patterns. The overall objective included both detecting instances of people and localizing the body joints. Each instance of a person was labeled with 17 joints. The COCO2017 dataset comprised 80 object categories and 250,000 instances of people with keypoints. The original images had different sizes but were resized to a pixel resolution of 384 × 288 and 320 × 320. The annotations of the training and validation sets were publicly benchmarked. During the evaluation, we followed the commonly used train2017/val2017 split.

This experiment was evaluated using OKS-based AP. The weights applied in our model were pre-trained using ImageNet [35]. The environmental settings were as follows. The experiment was conducted using the PyTorch library, on an Intel® Xeon® CPU E5-2690 v4 @ 2.60 GHz and an NVIDIA TITAN XP 1 GPU.

4.2. Evaluation metrics

We used OKS to evaluate the model performance for the COCO dataset. When measuring the accuracy, similar to the IoU applied during the object detection task, OKS was used in a keypoint estimation task, and it was calculated based on the distance between the predicted

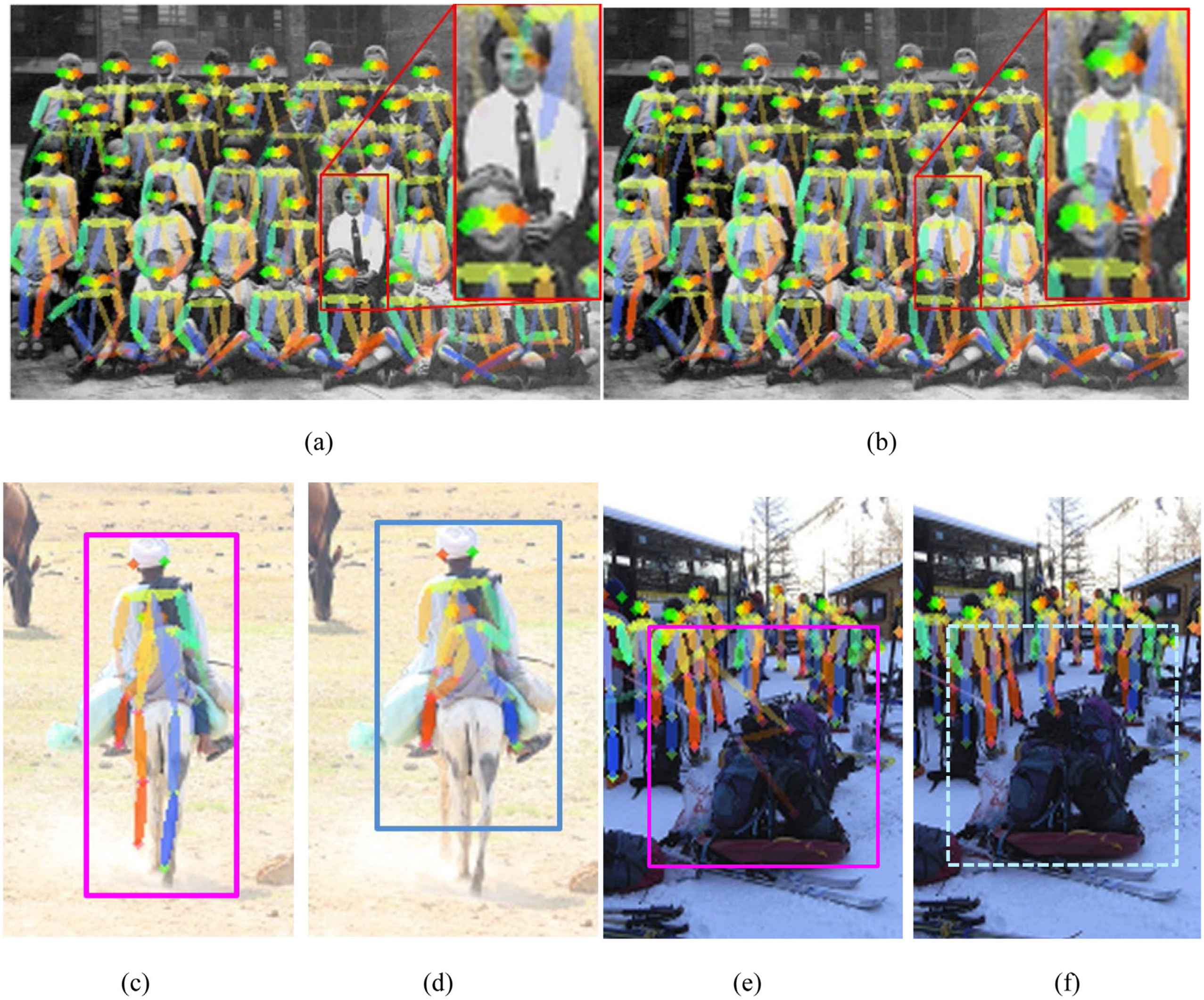

Fig. 7. Feedback-based loss output (right) and the original output (left) under complex conditions; the proposed method shows a higher performance: (a), (c), and (e) are the results of the original model; (b), (d), and (f) are the results when the proposed method was applied. The dotted box means that no unnecessary box has been detected.

points and the ground-truth points normalized by the scale of a person. This produced an accuracy score that can be used to measure the closeness between the two poses. In particular, each keypoint from the different body parts was weighted distinctively while measuring the distances. For example, the area of the eyes in the images was more prominent than those of the shoulders or hips, and therefore, the keypoints of the eyes should be strictly estimated. Using the typical method, the AP was calculated separately according to the minimum OKS value. Standard average precision and recall scores were as follows: AP 0.50 (AP at OKS = 0.50), AP 0.65, AP 0.75, AP 0.95, AP (the mean of AP scores at OKS = 0.50, 0.55, . . . , 0.90, 0.95). AP^M was for medium objects, and AP^L was for large objects.

4.3. Experiments with changing parameters

In this study, learning and evaluation were conducted while changing each parameter, which is presented along with the optimal value (Table 1). In addition, we found that the optimal hyperparameters obtained were similar in terms of accuracy even when they were increasing. To obtain reliable results, we averaged the results of 10 tests for each set of hyperparameters. The numerical changes for each of the parameters used in the test are illustrated in Fig. 5. The feedback-based loss function was configured to rapidly increase the loss value when the OKS value was less than 0.50. In addition, the penalty was obtained more rapidly as the value of β increased, as shown in the left part of Fig. 5. The right part of Fig. 5 shows a graph of the three hyperparameters of the object bounding box expansion. Because the number of persons on the x-axis is always 1 or more when detected, α has a minimum value of μ.

4.4. Ablation experiments with the addition of models

We tested each base model in our environment as shown in Table 2. The effect of the added model on the final performance can be seen through the model tested in our experimental environment and the results of testing each additional model separately. Among the models belonging to the top-down method, we conducted four experiments on Rmpe [26], HRNet-W48 [20,25], and DARK(+HRNet-W48) [35] models, respectively. The first is the result of the model we reproduced in our environment. The second is a model that only applies feedback-based loss optimization. The third is the model to which only bounding box expansion is applied, and the fourth is to show the performance of the model to which both methods are applied.

In addition, as shown in Table 4, we separately examined the performance of the object detection model used at the beginning of the model. The experimental data we used here are Hrnet-w48 + final model experiments in Table 2. These experimental results further demonstrate the single performance of a trained human detector. Additionally, it

Table 3
Comparison between models according to the number of people on COCO2017 keypoint validation dataset.

Methods	Number of people	Number of images	Accuracy				
			AP	AP 0.50	Ap 0.65	AP 0.75	AP 0.95
	1	1098	91.8	96.5	93.2	91.1	81.1
HRNet W48	2	454	88.4	92.3	87.3	89.0	80.5
DARK	3	251	75.9	90.1	85.4	88.7	77.2
	4+	543	73.3	85.6	80.1	82.3	72.7
Ours:	1	1098	93.6	98.2	93.8	91.3	81.2
HRNet W48	2	454	89.1	93.5	88.1	89.2	80.1
DARK +	3	251	75.8	91.3	85.7	88.7	77.0
Final model	4+	543	73.1	85.7	80.2	82.2	72.7

Table 4
Comparison of human detector performance and final keypoint estimator performance.

Human detector	Input size	Human detection accuracy (AP)	Keypoint estimation accuracy (AP)
Ground truth (GT)	–	–	76.8
Yolov3 416 [25]	416 × 416	55.5	73.9
Yolov3 416 + Ours	416 × 416	55.2	74.1

can be compared with the experimental results of the final model. From these experimental results, we can see that our model optimized the human detector for the final performance rather than a single performance.

4.5. Comparison with the latest technology

We compared our method to the latest representative MPPE models: PifPaf [31], SPM [32], Higher HRNet [33] and Simple Baseline [34]. Table 2 shows the accuracies of the latest methods and our proposed method when using the COCO val2017 dataset. First, we checked the accuracies of the baseline MPPEs with different input sizes. In addition, we checked and compared the accuracy of the proposed model with the same input size. As a result, we confirmed the effect of the two models, which are used for modifying the proposed object detector, on the pose estimator.

As shown in Table 2, first, we evaluated the results of each baseline model, the results of the two proposed methods, and the combined final model. We also compared the results of the top-down based model with the results of other comparison models. One thing to consider in this comparison is that while performance can be improved in the form of an add-on to the existing top-down model, it often performs worse than the more recent bottom-up approach. Both approaches are studied in separate ways because each has advantages in different situations. In addition, the feedback-based loss optimization method did not change the number of parameters of the models. However, it has been confirmed that training time can be increased up to 2.2 times per epoch. The time required for learning was increased from about 25 min per epoch to about 53 min based on HRNet -W48. Additionally, in terms of Rmpe, it increased from about 8 min per epoch to 17 min. Conversely, the detection expander did not affect training, but took an average of 10% more estimation time.

When the proposed method was applied, the accuracy at a high OKS score was higher than that obtained with the existing methods. This means that the proposed method achieves better results when a detailed pose estimation is required. The second method proposed herein was bounding box expansion. When the proposed method was applied, the overall accuracy was improved, particularly under low OKS scores. This result indicates that the number of objects approximately detected was more when the proposed method was applied.

Table 3 shows the number of people included in the images in the COCO2017 validation dataset. In addition, as shown in Table 3, the detection expander method is reasonable because many images contain only one or a few people. The images shown in Fig. 6(a) and (e) are the results of the original object detector. Fig. 6(b) and (f) show the results of applying the proposed bounding box expansion method to the object detector. Fig. 6(c) and (g) present the results of the poses estimated using the original object detector. In addition, Fig. 6(d) and (h) show the results of the pose estimator when applying the proposed method. In Fig. 6(d), more surrounding information can be seen than that in Fig. 6(c). As a result, the positions of the elbow and ankle of the leaning person are accurately estimated.

Fig. 7 shows the results of applying the feedback-based loss optimization method. Fig. 7(a), (c), and (e) show the results of the original model, and Fig. 7(b), (d), and (f) are the resulting images when the proposed method was applied. Unlike the image in Fig. 7(a), in Fig. 7(b), a person who was missed out in a complex image was detected. This means that the images were dense and complex, and in the case of multiple people appearing, the proposed method yields a higher performance. In Fig. 7(c), the leg of a horse was recognized as a human leg owing to an incorrect boundary box. In Fig. 7(d), the pose suitability of the bounding box was increased, and thus, the estimation result was more accurate. In Fig. 7(e), an unnecessary bounding box was added, and the baggage was incorrectly recognized as a person. When the proposed method was applied to the image, as shown in Fig. 7(f), unnecessary bounding boxes were not detected.

Overall, our model improved the accuracy compared to the baseline. In particular, the bounding box expansion model shows an accuracy improvement of 0.6 in terms of HRNet-W48 (at AP 0.50). In addition, the feedback-based loss optimization shows an accuracy improvement of 1.2 in terms of Rmpe (at AP 0.75). These results indicate that updating the object detector using the results of the pose estimator can improve the two-stage pose estimator results.

4.6. Failure cases

Fig. 8 shows some failure cases. First, cases in which large portions of an image are obscured by an object cannot be handled (e.g., Fig. 8(a)). If a human object detector misses detecting a person, the poses of the person will also go undetected (e.g., Fig. 8(b)). Additionally, in rare cases, the object detector may not be able to detect a person. For example, some poses that rarely occur may not be partially detected (e.g., Fig. 8(c)). Finally, our method is ineffective if only a part of a person appears in the image (e.g., Fig. 8(d)).

5. Conclusion

In this paper, we proposed a novel methodology for re-calibrating and optimizing object detectors for MPPE. First, we presented a feedback-based loss optimization that uses keypoint prediction accuracy as a penalty for the bounding box detectors. This method decreases the low-precision keypoint prediction probability but increases the high-precision keypoint prediction probability. Second, we proposed a

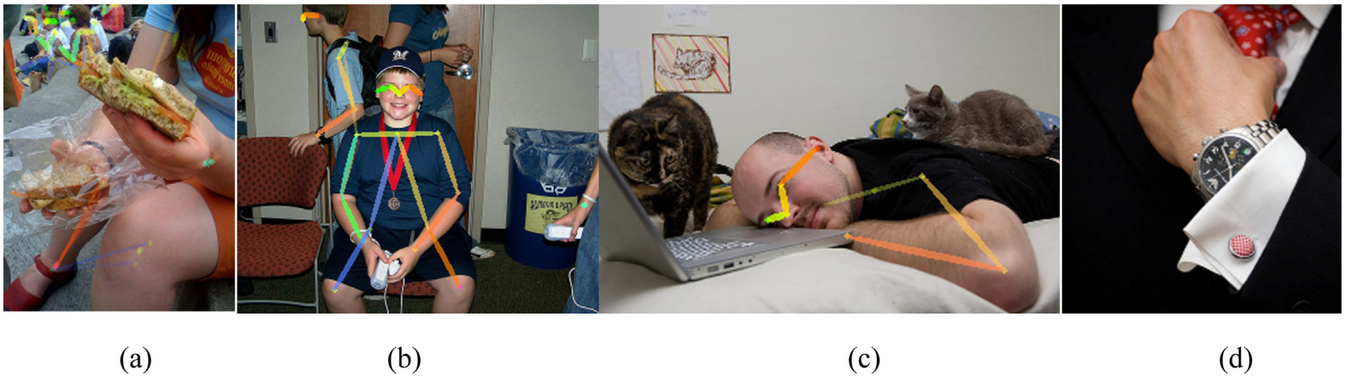

Fig. 8. Example cases where our models failed: when (a) an object obscures a large portion of the image, (b) the human object detector's result is undetected, (c) rare poses occur, and (d) only a part of a person appears in the image.

bounding box expander that appropriately extends the detected bounding box by taking into account the detected location and people within the detected location. The proposed bounding box expander reduces the feature loss during the keypoint estimation step. The experimental results showed that the proposed loss function and bounding box expansion method can perform real-time operations at 11 fps, which is similar to the performance of baseline models. In addition, the proposed model achieved an accuracy of 74.2 AP, which is higher than the latest model including the human detector used in the experiment.

CRediT authorship contribution statement

Jaeseo Park: Conception and design of study, Acquisition of data, Analysis and/or interpretation of data, Writing - original draft, Writing - review & editing. **Junho Heo:** Conception and design of study, Acquisition of data, Analysis and/or interpretation of data, Writing - original draft, Writing - review & editing. **Suk-Ju Kang:** Conception and design of study, Acquisition of data, Analysis and/or interpretation of data, Writing - original draft, Writing - review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

Funding: This research was supported by the Korea Agency for Infrastructure Technology Advancement (KAIA) grant funded by the Ministry of the Interior and Safety (Grant 21PQWO-B153358-03), and the Ministry of Science and ICT (MSIT), Korea, under the Information Technology Research Center (ITRC) support program (IITP-2021-2018-0-01421) supervised by the Institute for Information & Communications Technology Promotion (IITP), and Convergent Technology R&D Program for Human Augmentation through the National Research Foundation of Korea (NRF) funded by Ministry of Science and ICT (2020M3C1B8081320). All authors approved the version of the manuscript to be published.

References

[1] M.A. Fischler, R.A. Elschlager, The representation and matching of pictorial structures, IEEE Trans. Comput. C-22 (1) (2006) 67–92.
[2] M. Andriluka, S. Roth, B. Schiele, Pictorial structures revisited: People detection and articulated pose estimation, Comput. Vis. Pattern Recognit. (2009) 1014–1021.
[3] X. Chen, A. Yuille, Articulated pose estimation by a graphical model with image dependent pairwise relations, Adv. Neural Inf. Process. Syst. (2014) 1736–1744.
[4] B. Sapp, B. Taskar, Modec: Multimodal decomposable models for human pose estimation, Comput. Vis. Pattern Recognit. (2013) 3674–3681.
[5] Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, Gradientbased learning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324.
[6] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, The IEEE Conf. Comput. Vis. Pattern Recognit. (2016) 770–778.
[7] Z. Cao, T. Simon, S.E. Wei, Sheikh, Realtime multi-person 2d pose estimation using part affinity fields, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2017) 7291–7299.
[8] S.E. Wei, V. Ramakrishna, T. Kanade, Y. Sheikh, Convolutional pose machines, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2016) 4724–4732.
[9] A. Newell, K. Yang, J. Deng, Stacked hourglass networks for human pose estimation, Eur. Conf. Comput. Vis. (2016) 483–499.
[10] G. Papandreou, T. Zhu, N. Kanazawa, A. Toshev, J. Tompson, C. Bregler, K. Murphy, Towards accurate multiperson pose estimation in the wild, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2017) 4903–4911.
[11] W. Ouyang, X. Chu, X. Wang, Multi-source deep learning for human pose estimation, Comput. Vis. Pattern Recognit. (2014) 2329–2336.
[12] A. Toshev, C. Szegedy, Deeppose: Human pose estimation via deep neural networks, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2014) 1653–1660.
[13] V. Belagiannis, A. Zisserman, Recurrent human pose estimation, in: 12th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG), 2017, pp. 468-475.
[14] A. Bulat, G. Tzimiropoulos, Human pose estimation via convolutional part heatmap regression, Eur. Conf. Comput. Vis. (2016) 717–732.
[15] Z. Cao, T. Simon, S.E. Wei, Y. Sheikh, OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields, 2018, arXiv preprint arXiv:1812.08008.
[16] L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P.V. Gehler, B. Schiele, Deepcut: Joint subset partition and labeling for multi person pose estimation, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2016) 4929–4937.
[17] K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask r-cnn, Proc. IEEE Int. Conf. Comput. Vis. (2017) 2961–2969.
[18] F. Zhang, X. Zhu, H. Dai, M. Ye, C. Zhu, Distribution-aware coordinate representation for human pose estimation, 2019, arXiv preprint arXiv:1910.06278.
[19] K. Sun, B. Xiao, D. Liu, J. Wang, Deep high-resolution representation learning for human pose estimation, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2019) 5693–5703.
[20] Wang. J., Sun. K., Cheng. T., Jiang. B., Deng. C., Zhao. Y., Xiao. B., et al., Deep high-resolution representation learning for visual recognition, IEEE Trans. Pattern Anal. Mach. Intell. (2020).
[21] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, Adv. Neural Inf. Process. Syst. (2015) 91–99.
[22] A. Güler, Rıza, N. Neverova, I. Kokkinos, Densepose: Dense human pose estimation in the wild, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2018) 7297–7306.
[23] William MCNALLY, et al., EvoPose2D: Pushing the boundaries of 2D human pose estimation using neuroevolution, 2020, arXiv preprint arXiv:2011.08446.
[24] Junjie Huang, et al., AID: Pushing The performance boundary of human pose estimation with information dropping augmentation, 2020, arXiv preprint arXiv:2008.07139.
[25] Joseph Redmon, A. Farhadi, Yolov3: An Incremental improvement, 2018, arXiv preprint arXiv:1804.02767.
[26] H.S. Fang, S. Xie, Y.W. Tai, C. Lu, Rmpe: Regional Multi-person pose estimation, 2018, arXiv preprint arXiv:1612.00137v5.
[27] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, Real-Time Object Detection, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2016) 779–788.
[28] D. Kim, S. Park, D. Kang, J. Paik, Real-time robust object detection using an adjacent feature fusion-based single shot multibox detector, IEIE Trans. Smart Process. Comput. 9 (1) (2020) 22–27.

[29] Z.Q. Zhao, P. Zheng, S.T. Xu, X. Wu, Object detection with deep learning: A review, IEEE Trans. Neural Netw. Learn. Syst. 30 (11) (2019) 3212–3232.

[30] T.Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, C.L. Zitnick, et al., Microsoft coco: Common objects in context, Eur. Conf. Comput. Vis. (2014) 740–755.

[31] Sven Kreiss, Lorenzo Bertoni, Alexandre Alahi, Pifpaf: Composite fields for human pose estimation, Proc. IEEE Int. Conf. Comput. Vis. (2019) 11977–11986.

[32] Nie. X., Feng. J., Zhang. J., Yan. S., Single-stage multi-person pose machines, Proc. IEEE Int. Conf. Comput. Vis. (2019) 6951–6960.

[33] Cheng. B., Xiao. B., Wang. J., Shi. H., Huang. T.S., Zhang. L., HigherHRNet: Scale-aware representation learning for bottom-up human pose estimation, Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (2020) 5386–5395.

[34] Bin Xiao, Haiping Wu, Yichen Wei, Simple baselines for human pose estimation and tracking, Proc. Eur. Conf. Comput. Vis. (2018) 466–481.

[35] J. Deng, W. Dong, R. Socher, L.J. Li, K. Li, F.-F. Li, ImageNet: A large-scale hierarchical image database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 248–255.

[36] F. Zhang, X. Zhu, H. Dai, M. Ye, C. Zhu, Distribution-aware coordinate representation for human pose estimation, Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (2020) 7093–7102.