Multi-scale Aggregation R-CNN for 2D Multi-person Pose Estimation

Gyeongsik Moon
Department of ECE, ASRI
Seoul National University
mks0601@snu.ac.kr

Ju Yong Chang
Department of EI
Kwangwoon University
juyong.chang@gmail.com

Kyoung Mu Lee
Department of ECE, ASRI
Seoul National University
kyoungmu@snu.ac.kr

Abstract

Multi-person pose estimation from a 2D image is challenging because it requires not only keypoint localization but also human detection. In state-of-the-art top-down methods, multi-scale information is a crucial factor for the accurate pose estimation because it contains both of local information around the keypoints and global information of the entire person. Although multi-scale information allows these methods to achieve the state-of-the-art performance, the top-down methods still require a huge amount of computation because they need to use an additional human detector to feed the cropped human image to their pose estimation model. To effectively utilize multi-scale information with the smaller computation, we propose a multi-scale aggregation R-CNN (MSA R-CNN). It consists of multi-scale RoIAlign block (MS-RoIAlign) and multi-scale keypoint head network (MS-KpsNet) which are designed to effectively utilize multi-scale information. Also, in contrast to previous top-down methods, the MSA R-CNN performs human detection and keypoint localization in a single model, which results in reduced computation. The proposed model achieved the best performance among single model-based methods and its results are comparable to those of separated model-based methods with a smaller amount of computation on the publicly available 2D multi-person keypoint localization dataset.

1. Introduction

Localizing semantic keypoints of an instance such as a human body or hand is an essential technique for action recognition or human-computer interaction. It has been studied for decades in computer vision community and has attracted considerable research interest.

Recently, many methods [11, 5, 13, 2, 14, 22, 20, 24] utilize deep convolutional neural networks (CNNs) and achieved noticeable performance improvement. Although these methods have progressed considerably, they still suffer from occluded or invisible keypoints, crowded background, and high computational complexity.

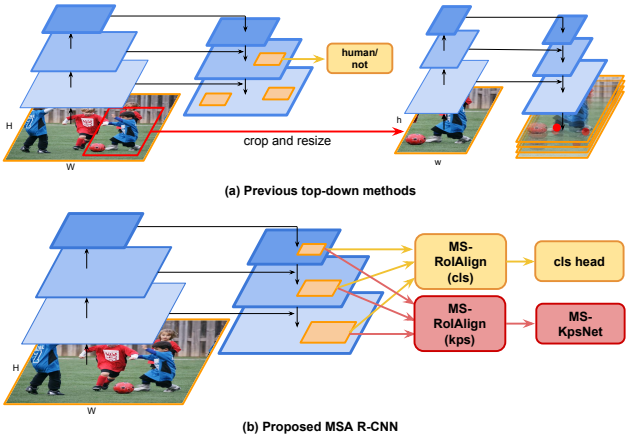

Figure 1: Overall pipeline comparison with the previous top-down methods (a) and the proposed method (b). Most of the top-down approaches use two separated deep networks for multi-person pose estimation. The first model is a human detector (*i.e.*, left part of (a)) and the other is a pose estimation model (*i.e.*, right part of (a)). In contrast, in (b), the human detector (*i.e.*, cls head) and pose estimation network (*i.e.*, MS-KpsNet) are combined into a single model and share most of the feature maps.

In the previous top-down methods, the use of multi-scale information is crucial in performance improvement. Newell *et al*. [21] and Chen *et al*. [5] used downsampling and upsampling layers with skip connections. This network architecture (*i.e.*, U-net structure) is simple and effective. Huang *et al*. [13] aggregated multi-scale information by concatenating feature maps from multiple scale spaces. Although these multi-scale approaches exhibit state-of-the-art accuracy, they require a huge amount of computation because they need to use an additional human detector to feed the cropped human image to their model. Considering that both of the recent state-of-the-art object detectors [25, 11] and keypoint localization networks [5, 13, 22] are primarily based on the very deep backbone networks [12, 34], the total amount of computation is very large.

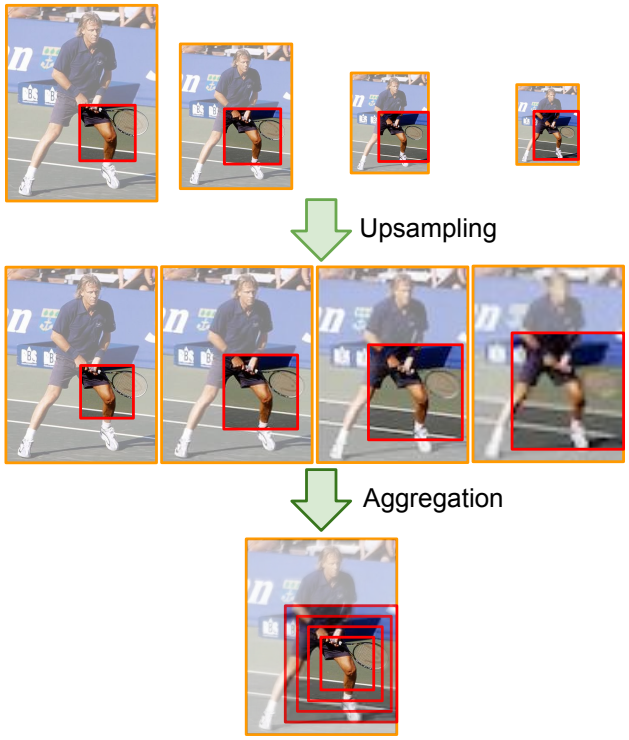

Figure 2: The MSA R-CNN extracts multi-scale information from downsampled and upsampled feature maps and aggregates the information by using MS-RoIAlign and MS-KpsNet. The orange and red boxes denote the extracted feature maps of the human and the receptive fields of convolutional layers, respectively. We take an example of the left knee area.

By contrast, Mask R-CNN [11] learns human detection and keypoint localization in a single model that can be trained in an end-to-end manner. Based on the shared feature maps, two small separated head networks for human/non-human classification and keypoint localization are jointly learned to minimize the weighted sum of loss functions. However, this method does not fully utilize multi-scale information which is a bottleneck to the accurate keypoint localization. Specifically, RoIAlign [11] extracts a feature of each proposal from a single-scale feature map by considering the size of each proposal. The small proposals are extracted from a fine-scaled feature map, while the large proposals from a coarse-scaled feature map. However, because *each* proposal is from a *single-scale* feature map, RoIAlign fails to fully exploit multi-scale information. Also, the keypoint head network consists of several sequentially added convolutional layers. As this design gradually increases receptive field size, the output feature would mainly focus on global information rather than local information. This makes it hard to aggregate multi-scale information.

To remedy the heavy computation in the previous top-down methods [5, 13, 24] and the lack of multi-scale information in the Mask R-CNN [11], we propose a multi-scale aggregation R-CNN (MSA R-CNN). The MSA R-CNN crops and resizes human bounding box proposals from feature maps instead of an input image as shown in Figure 1. This property enables the MSA R-CNN to share feature maps for human detection and keypoint localization, which results in considerably reduced computation. Also, to exploit multi-scale information more effectively, we propose multi-scale RoIAlign block (MS-RoIAlign) and multi-scale keypoint head network (MS-KpsNet). In contrast to the original RoIAlign, the MS-RoIAlign obtains human proposals from multi-scale feature maps instead of a single feature map and aggregates them. It enables the model to exploit various scales of the feature maps which is helpful for the final prediction. Also, the MS-KpsNet obtains human proposals from the MS-RoIAlign and estimates heatmaps for each keypoint by utilizing multi-scale information. The proposed MS-KpsNet consists of downsampling and upsampling layers with residual skip connections which help incorporate local- and global-scale information. To summarize, both of the MS-RoIAlign and MS-KpsNet try to extract and aggregate multi-scale information as in Figure 2.

We validated the usefulness of the MS-RoIAlign and MS-KpsNet on the MS COCO keypoint detection dataset [18]. The experimental results show that the proposed items (*i.e.*, MS-RoIAlign and MS-KpsNet) bring large performance improvement. Our model outperforms all single model-based methods and achieves comparable results to those of separated model-based methods but with less computation on a challenging benchmark [18].

Our contributions can be summarized as follows:

- The MSA R-CNN reduces a large amount of computation compared with other top-down methods by combining human detection and keypoint localization in a single model.
- The MS-RoIAlign and MS-KpsNet effectively utilize multi-scale information, thereby enhancing performance.
- Our model achieved the best performance among single model-based methods and comparable results to those of separated model-based methods on the MS COCO keypoint detection dataset [18].

2. Related works

The proposed method is closely related to the following two tracks. In this paper, we mainly focus on methods based on the CNN.

Single-person pose estimation. Toshev *et al.* [31] directly estimated the Cartesian coordinates of body joints

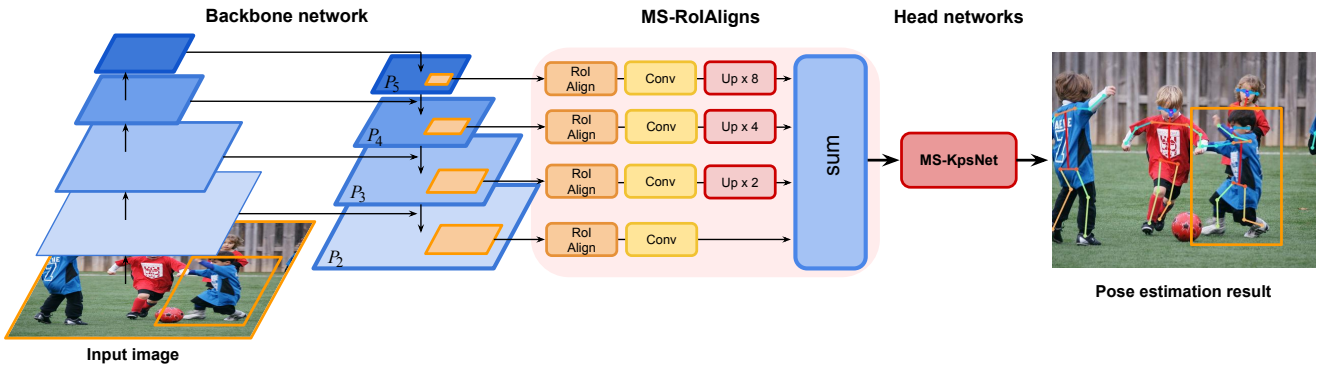

Figure 3: Overall pipeline of the proposed method. The input image that contains multiple humans is fed to the backbone network. After the backbone network generates human bounding box proposals, the features of the proposals are extracted by the MS-RoIAlign from the multiple feature maps. The extracted features are aggregated and passed to the head networks in a parallel manner. We exclude the MS-RoIAlign and head network of the classification from the figure, and only one proposal with orange rectangle is drawn for simplicity.

by using a multi-stage deep network and obtained remarkable performance. Tompson *et al*. [30] estimated the per-pixel likelihood for each joint by using CNN and used it as the unary term in an external graphical model to accurately estimate joint positions. Liu *et al*. [32] utilized multiple stages of refinement to enlarge receptive fields. Newell *et al*. [21] proposed a stacked U-Net structure model (*i.e.*, hourglass structure) to exploit information from multiple scales. Bulat and Tzimiropoulos [1] adopted a detection subnetwork to help the regression subnetwork accurately localize body joints. Carreria *et al*. [3] proposed an iterative error feedback-based human pose estimation system. It is supervised to progressively refine the initial pose to the groundtruth pose. Chu *et al*. [6] enhanced the stacked hourglass network [21] by incorporating multi-context attention mechanism. Chen *et al*. [4] also improved the hourglass network [21] with the adversarial loss to generate plausible poses.

Multi-person pose estimation. There are two streams in the multi-person pose estimation. The first one, top-down approach, relies on a human detector which predicts bounding boxes of humans. The detected human is cropped and fed to the pose estimation network. The second one, bottom-up approach, localizes all the human body keypoints in an input image and groups them using proposed clustering algorithms of each work.

[11, 5, 13, 22, 33, 19] are based on the top-down approach. Papandreou *et al*. [22] estimated heatmaps and offsets for each joint. The offsets are defined as vectors toward the groundtruth joint location from each tensor grid. He *et al*. [11] proposed Mask R-CNN which can perform human detection and keypoint localization in a single model. It extracts human features from a feature map instead of an input image by using RoIAlign. Chen *et al*. [5] used a coarse-to-fine approach and designed a network called cascaded pyramid network (CPN) which consists of GlobalNet and RefineNet. The GlobalNet is U-Net shaped model and supervised to estimate heatmaps for each keypoint from each scale of a feature map. The RefineNet is designed to refine the localization output from the GlobalNet by focusing on hard keypoints. Xiao *et al*. [33] proposed a straightforward architecture-based human pose estimation model.

[2, 14, 20, 24, 16] are based on the bottom-up approach. DeepCut [24] assigned the detected keypoints to different persons in an image by formulating the assignment problem as an integer linear program. DeeperCut [24] improves the DeepCut [24] by introducing image-conditioned pairwise terms. Cao *et al*. [2] proposed part affinity fields (PAFs) that models the relationship between human body keypoints and assembled the localized keypoints using the estimated PAFs. Newell *et al*. [20] introduced a pixel-wise tag value to assign localized keypoints to a certain human. Kocabas *et al*. [16] proposed a pose residual network for assigning detected keypoints to each person.

3. Overview of the proposed model

The proposed MSA R-CNN has three components. The first is a single backbone network for shared feature extraction. The second component is separated into two MS-RoIAligns for human/non-human classification and keypoint localization. The outputs of MS-RoIAligns are fed to two small head networks (*i.e.*, classification head network and MS-KpsNet) which are the third component of our system. The backbone network extracts deep features, and each MS-RoIAlign passes these features to the corresponding head network. The classification head network predicts whether a proposal is human or not, and the MS-KpsNet estimates heatmaps for each joint. The overall pipeline is visualized in Figure 3.

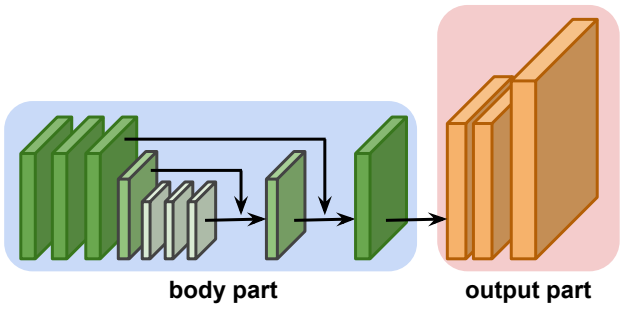

Figure 4: Architecture of the MS-KpsNet. It consists of a body and output parts. The feature map in the body part passes through convolutional, downsampling and upsampling layers. In the output part, the feature map from the body part is upsampled by a deconvolutional layer and bilinear interpolation is applied for accurate estimation. The loss is calculated on the four times upsampled RoI which is the last feature map of the output part.

4. Backbone network for shared features

The feature pyramid network (FPN) [17] is adopted as the backbone network. The FPN extracts deep features using ResNet [12] or ResNeXt [34] and gradually upsamples the features. Each upsampled feature is summed by lateral connections with the feature map in the same scale space from the front part of the network. This upsampling with skip connection architecture is widely used for dense prediction such as segmentation [27] and keypoint localization [21] because it can provide more semantic information to fine-scale feature maps. Following [25], the backbone network is supervised to generate human bounding box proposals from an input image by using a binary cross entropy loss for each sampled feature map grid and a smooth L1 loss to refine the bounding box coordinates.

5. Multi-scale RoIAlign block (MS-RoIAlign)

The MS-RoIAlign passes the extracted human feature from the backbone network to the corresponding head network.

The original RoIAlign [11] extracts human proposal features from a single feature map. The feature map is selected among several scales according to the size of the proposal [17]. The original method assigns small and large proposals to large feature maps (fine-scale, low-level feature maps) and small feature maps (coarse-scale, high-level feature maps), respectively. However, this straightforward assignment strategy can result in sub-optimal performance. For example, two proposals that have almost the same area can be assigned to two different feature maps. Such an assignment can make learning unstable because the proposals have similar areas. Hence, we consider feature maps from the entire-scale space instead of a single-scale feature map. Another disadvantage of the original RoIAlign is that other levels of feature maps are discarded. Exploiting multi-level feature maps provides more information than exploiting only a single feature map. The low-level features contain detailed local information, which results in high localization accuracy in the fine-scale space. Furthermore, the high-level features have rich semantic information resulting from the large receptive field size in the coarse-scale space. Compared with the existing RoI assignment strategy [17], the proposed MS-RoIAlign can utilize all information from multi-level feature maps.

The pipeline of the MS-RoIAlign is visualized in Figure 3. The MS-RoIAlign extracts ($2^{n+3}\times2^{n+3}$, $2^{n+2}\times2^{n+2}$, $2^{n+1}\times2^{n+1}$, $2^{n}\times2^{n}$)-sized RoIs from upsampled feature maps (P_2, P_3, P_4, P_5) for each proposal. The extracted features go through convolutional layers followed by subsequent upsampling layers. The RoIs are resized to a fixed size (*i.e.*, $2^{n+3}\times2^{n+3}$) and aggregated by summation. Then, it is fed to the corresponding head network. This procedure lets the following head networks fully utilize the multi-scale features instead of narrowing the choice to a single-scale feature. The n is set to 0 for the classification and 1 for the keypoint localization to make the RoI sizes similar to those of the Mask R-CNN [11]. Except for the parameter n related to the size of the input RoI, the MS-RoIAligns of the two tasks have exactly the same architecture. The small difference in the RoI sizes of our method and Mask R-CNN [11] makes no difference in terms of the performance.

6. Multi-scale keypoint head network (MS-KpsNet)

The human proposal features extracted by the MS-RoIAlign are fed to the proposed MS-KpsNet which predicts heatmaps for each keypoint. To effectively utilize both of the local- and global-scale information, the MS-KpsNet is designed with downsampling and upsampling architectures and residual skip connections.

The architecture of the MS-KpsNet is presented in Figure 4. The MS-KpsNet starts from three consecutive convolutional layers and goes through two rounds of downsampling. Each downsampling layer is followed by a convolutional layer. The downsampled feature passes two convolutional layers and subsequently upsampled followed by a residual skip connection. The forward is finished after two rounds of upsampling and skip connection. Like the downsampling layers, a convolutional layer is added after each residual skip connection in the upsampling part. Max pooling with stride and kernel size of 2 is used for downsampling layers and nearest neighbor with a scale factor of 2 is used for upsampling layers. The skip connection is a single convolutional layer. All the convolutional layers

Methods	AP^{kps}	$AP^{kps}_{.50}$	$AP^{kps}_{.75}$	AP^{kps}_{M}	AP^{kps}_{L}	$AP^{bb(H)}$	$AP^{bb(H)}_{.50}$	$AP^{bb(H)}_{.75}$	$AP^{bb(H)}_{S}$	$AP^{bb(H)}_{M}$	$AP^{bb(H)}_{L}$
Baseline	64.1	86.4	69.3	58.9	72.7	52.7	82.3	57.4	35.6	60.5	68.7
+ Only from P_2	64.4	86.4	69.9	59.4	72.8	52.5	82.2	57.2	35.4	60.7	68.2
+ 1×1 conv output	64.7	86.3	70.4	59.6	73.2	52.4	82.5	56.9	35.2	60.6	68.0
+ MS-KpsNet	66.2	87.0	72.7	61.3	74.5	52.6	82.3	57.4	35.6	60.6	68.3
+ Longer training	66.5	87.5	72.5	61.5	75.0	53.4	82.8	58.3	36.1	61.5	69.3
+ MS-RoIAlign	67.4	87.7	73.5	62.1	76.0	54.8	83.4	60.1	37.4	62.8	71.2
+ Average of Top-2s	67.6	87.7	73.7	62.5	76.1	54.8	83.4	60.1	37.4	62.8	71.2
+ Test-time augmentation	**70.3**	**89.2**	**76.6**	**65.9**	**77.9**	**56.4**	**84.9**	**61.8**	**39.0**	**64.2**	**72.6**
	+6.2	**+2.8**	**+7.3**	**+7.0**	**+5.2**	**+3.7**	**+2.6**	**+4.4**	**+3.4**	**+3.7**	**+3.9**

Table 1: Effect of various settings in terms of the performance on the MS COCO validation set. $AP^{bb(H)}$ means the average precision of detection task for the human class only.

Aggregation	AP^{kps}	Num of params	Train mem
Sum	67.6	76.3M	10.8 GB
Concat	67.6	137.2M	14.4 GB

Table 2: Performance comparison of the MS-RoIAlign with different aggregation method. The AP is from the test result of the MS COCO validation set. The train mem indicates the required amount of GPU memory in the training stage.

have 3×3 kernels and are followed by the activation function (*i.e.*, ReLU). Cross-entropy loss function L is calculated after softmax normalization as follows:

$$L = -\frac{1}{N}\sum_{n=1}^{N}\sum_{i,j} H^*_n(i,j)\log H_n(i,j), \qquad (1)$$

where H^*_n and H_n are the groundtruth and estimated heatmaps with softmax applied for nth keypoint, respectively, and N denotes the number of keypoints. Groundtruth heatmap H^*_n is encoded as a one-hot representation.

7. Implementation details

Our model is based on the official Caffe2 [15] implementation of the Mask R-CNN [9]. Following the Mask R-CNN [11], human bounding box proposals are generated from an independently trained RPN [17, 25] for convenient ablation study and fair comparison. Note that it can be trained in an end-to-end manner and achieves slightly better results compared with the model trained from independently trained RPN [9].

Training. Our model is based on ResNet-50 [12] and all weights are initialized with a publicly released model pre-trained on the ImageNet dataset [28]. We adopt image-centric training [8]. For each image, we sample 512 RoIs with positive-to-negative ratio of 1:3. For data augmentation, the length of the short side of an image is randomly sampled between 640 and 800 pixels. Weight decay and momentum are set to 0.0001 and 0.9, respectively. As we used two GPUs that are smaller than that of the Mask R-CNN [11], we used the linear scaling rule [10] to set the learning rate and number of iterations according to the number of GPUs. Each GPU takes 2 images to generate a mini-batch. For the classification head network, we used the same loss function (i.e., binary cross-entropy) and architecture (i.e., two fully-connected layers) as the Mask R-CNN [11].

Inference. At test time, the extracted RoI bounding boxes pass the classification head network and the estimated bounding box refinement vector refines the coordinates of the bounding boxes. Then, the refined bounding boxes pass the MS-KpsNet, which differs from the parallel computation used in training. This sequential prediction speeds up inference and improves accuracy due to the use of fewer and more accurate RoIs. The predicted heatmaps for each body keypoint are resized to the original RoIs and the position of the highest response for each keypoint is identified and warped to the final result of our model.

All the hyper-parameters are adopted from Mask R-CNN [11] and FPN [17] without any fine-tuning.

8. Experiment

8.1. Dataset and evaluation metric

The proposed model is trained on the MS COCO [18] training set which includes 57K images and 150K person instances. The validation is performed on the MS COCO validation set which includes 5K images and testing is conducted on the test-dev set that includes 20K images. Following the public benchmark, we used the object keypoint similarity (OKS) [26] based mAP as an evaluation metric. The OKS defines the similarity between the coordinates of two human body keypoints which is similar to intersection over union in object detection.

8.2. Ablation study

We trained our model on the MS COCO training set and validated the proposed components on the MS COCO validation set.

Multi-scale aggregation network. To demonstrate the

Methods	Backbone	AP^{kps}	$AP^{kps}_{.50}$	$AP^{kps}_{.75}$	AP^{kps}_{M}	AP^{kps}_{L}	AR^{kps}	$AR^{kps}_{.50}$	$AR^{kps}_{.75}$	AR^{kps}_{M}	AR^{kps}_{L}
Separated model-based methods											
RMPE [7]	-	61.0	82.9	68.8	57.9	66.5	-	-	-	-	-
G-RMI [22]	ResNet-101	64.9	85.5	71.3	62.3	70.0	69.7	88.7	75.5	64.4	77.1
CPN [5]	ResNet-Inception	72.1	**91.4**	**80.0**	68.7	**77.2**	**78.5**	**95.1**	**85.3**	**74.2**	**84.3**
CFN [13]	Inception v2	**72.6**	86.1	69.7	**78.3**	64.1	-	-	-	-	-
Single model-based methods											
CMU-Pose [2]	-	61.8	84.9	67.5	57.1	68.2	66.5	87.2	71.8	60.6	74.6
Mask R-CNN [11]	ResNet-50-FPN	63.1	87.3	68.7	57.8	71.4	-	-	-	-	-
AE [20]	-	65.5	86.8	72.3	60.6	72.6	70.2	89.5	76.0	64.6	78.1
MSA R-CNN (Ours)	ResNet-50-FPN	**68.2**	**89.7**	**75.0**	**63.8**	**75.6**	**74.4**	**93.4**	**80.3**	**69.2**	**81.5**

Table 3: Comparison with the state-of-the-art methods on the MS COCO test-dev set. Methods that involve extra training data or use ensemble technique are excluded.

Method	AP_{kps}	Running time	Train mem
CPN-50	67.3	0.10 + 0.21	7.9 + 10.5 GB
Mask R-CNN	63.1	0.17	7.7 GB
Mask R-CNN+	67.0	0.31	14.2 GB
MSA R-CNN (Ours)	67.6	0.21	10.8 GB

Table 4: Computational complexity comparison with the state-of-the-art top-down methods. The AP is from the test result on the MS COCO validation set. The running time is the number of seconds required to process an image and the train mem indicates the amount of the GPU memory consumption in the training stage. For CPN-50, the former and latter results are from the human detector and the pose estimation model, respectively.

validity of the multi-scale aggregation, we compared the performances of the baseline model [11] and the proposed MSA R-CNN in Table 1.

1) Baseline model. We employed ResNet-50-based Mask R-CNN [11] as the baseline model.

2) Only from P_2. When all human bounding box proposals are extracted from the finest-scale feature map (P_2), not from the assigned feature map according to size [17], the performance is slightly improved. This may be because the keypoint localization task prefers features from large upsampled feature maps. Although the extracted RoI size is the same regardless of whether it is from the P_2 to the P_5, the detailed local and fine-scaled information from the P_2 is helpful for accurate keypoint localization.

3) 1×1 conv output. In the output part of the original keypoint head network, we used a deconv layer followed by an 1×1 conv layer to generate heatmaps for each joint instead of a single deconv layer. This is to separate the two tasks, which upsample the feature map and estimate the heatmap.

4) MS-KpsNet. When the proposed MS-KpsNet is introduced, the performance increases by 1.5 AP. This shows the usefulness of the MS-KpsNet that is designed to effectively utilize multi-scale information.

5) Longer training. For stable convergence, we scaled the training schedule by approximately 1.44 times, which slightly improves performance.

6) MS-RoIAlign. The MS-RoIAlign increases performance by 0.9 AP which shows utilizing multi-scale features is better than relying on a single-scale feature.

7) Average of Top-2s. To improve performance of the keypoint localization at high precision thresholds, we select top-2 grid with the highest probability from the estimated heatmap. Then, the weighted average of the locations of the selected grids based on their probability becomes the final location of each keypoint.

8) Test-time augmentation. The multi-scale test-time augmentation is commonly used to boost the performance [20, 2]. It averages heatmaps from multiple sizes of an input image, which makes the model robust to scale variations.

All the proposed methods obtain 6.2 AP improvement compared with the baseline model.

Aggregation method. We explore the best aggregation method in the MS-RoIAlign in terms of the performance and computational complexity. We compared our aggregation method (*i.e.*, summation) with concatenation which is used in CFN [13]. When concatenation is used, the upsampled RoIs are concatenated along the channel dimension. The first three feature map dimension of the MS-KpsNet are changed to 1024, 1024, and 512 in response to the increased number of channels. As Table 2 shows, there is a marginal performance difference between concatenation and summation although concatenation requires more parameters and consumes more GPU memory in the training stage. Therefore, we used summation as the aggregation method in the MS-RoIAlign.

8.3. Comparison with the state-of-the-art methods

We compared the performance of the MSA R-CNN on the MS COCO [18] test-dev set with that of recent state-of-the-art methods including RMPE [7], CMU-Pose [2], Mask

R-CNN [11], Associative Embedding (AE) [20], CFN [13], G-RMI [22], and CPN [5]. Table 3 shows the performance comparison.

Our MSA R-CNN outperforms all the single model-based methods. We additionally tried to compare the proposed MSA R-CNN with a recently introduced single model-based method, MultiPoseNet [16]. As they only reported the performance using ensemble on the test-dev set, we compare our MSA R-CNN with the MultiPoseNet [16] on the validation set without ensemble and testing time augmentation. Our ResNet-50-based model achieves 67.6 mAP while their ResNet-50-based model achieves 62.3 mAP. Moreover, their model with deeper backbone network (*i.e.*, ResNet-101) achieves 63.9 mAP which is still lower than ours. This comparison clearly shows the proposed MSA R-CNN outperforms all the single model-based methods.

On the other hand, the proposed method performs slightly worse than recent state-of-the-art top-down methods [13, 5] that require an additional human detection model. As our model contains both of the human detector and keypoint localization network, a limit exists in the use of computational resources such as GPU memory, which poses a limitation in obtaining better performance. This prevents us from utilizing well-known factors for performance boosting such as a deeper backbone network [17].

By contrast, separated model-based methods train and test the human detector and pose estimation model separately. Accordingly, a computational resource limitation exists for each model and not the combined model. The increased computational resource limitation can be used for performance enhancement. For example, recent state-of-the-art top-down methods use very deep network-based human detectors [11, 25, 23], which consume a large amount of computation resource. The CPN used human detection results from the MegDet [23] which is trained on 128 GPUs. The MegDet [23] obtains 50.5 AP on the MS COCO [18] detection validation set for all classes whereas our baseline (*i.e.*, ResNet-50-based Mask R-CNN) obtains 37.3 AP. Moreover, their keypoint localization models not only can use very deep backbone networks including ResNeXt [34] and ResNet-Inception [29], but also can be designed with as highly sophisticated network architecture [13, 5].

Figure 5 shows the qualitative results of our MSA R-CNN on the MS COCO [18] keypoint detection test-dev set.

8.4. Computational complexity

We compared the accuracy and computational complexity of the proposed method with those of the Mask R-CNN, very deep backbone based-Mask R-CNN [11] (*i.e.*, Mask R-CNN+) and the basic model of the CPN [5] (*i.e.*, CPN-50) in Table 4. Among the separated model-based methods, we chose the CPN because it released the code and achieved top performance. The Mask R-CNN+ uses ResNeXt-101-FPN [34, 17] as a backbone network and CPN-50 is based on the ResNet-50 [12]. We use the same backbone based object detector with ours (*i.e.*, ResNet-50-FPN-based Mask R-CNN) as the human detector of the CPN-50 because the human detector code of the CPN (i.e., MegDet [23]) is unavailable. For a fair comparison, ensembling, and keypoint rescoring, and test time augmentation techniques are excluded.

As Table 4 shows, our method achieves the best accuracy with the least amount of computational resource in both of the training and testing stages compared with the Mask R-CNN+ and CPN-50. The CPN-50 requires 48% longer running time in the testing stage and 70% more GPU memory in the training stage to achieve similar accuracy with the MSA R-CNN. Considering that the CPN-50 is the simplest model of the CPN with a basic human detector, previous separated model-based systems require a huge amount of computation to achieve the state-of-the-art performance.

Furthermore, compared with the Mask R-CNN, the MSA R-CNN increases the computational complexity by approximately 30% whereas the Mask R-CNN+ increases it by around 80% in both of the training and testing stages. This result indicates that the proposed modules in the MSA R-CNN (*i.e.*, MS-RoIAlign and MS-KpsNet) efficiently increases accuracy compared with using deeper backbone network which is the most widely used strategy for accuracy improvement [11, 12, 17].

9. Conclusion

We proposed a novel and powerful network, MSA R-CNN, for 2D multi-person pose estimation. In contrast to previous top-down methods, the proposed method performs human detection and keypoint localization in a single model. This unified model allows us to save a large amount of computations compared with the separated model-based methods. Also, to effectively utilize multi-scale information, the MS-RoIAlign and MS-KpsNet are proposed, which extract multi-scale features and aggregate them. MS-RoIAlign and MS-KpsNet obtain remarkable performance improvements. Our method outperforms all the existing single model-based methods and achieved comparable results to those of the separated model-based methods on the challenging benchmark. Codes will be released for reproduction.

Acknowledgments

This work was supported by deep learning-based human pose estimation system development project from NAVER.

Figure 5: Qualitative results of our MSA R-CNN on the MS COCO test-dev dataset.

References

[1] Adrian Bulat and Georgios Tzimiropoulos. Human pose estimation via convolutional part heatmap regression. In *ECCV*, 2016.

[2] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. *CVPR*, 2017.

[3] Joao Carreira, Pulkit Agrawal, Katerina Fragkiadaki, and Jitendra Malik. Human pose estimation with iterative error feedback. In *CVPR*, 2016.

[4] Yu Chen, Chunhua Shen, Xiu-Shen Wei, Lingqiao Liu, and Jian Yang. Adversarial learning of structure-aware fully convolutional networks for landmark localization. In *ICCV*, 2017.

[5] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. *CVPR*, 2018.

[6] Xiao Chu, Wei Yang, Wanli Ouyang, Cheng Ma, Alan L Yuille, and Xiaogang Wang. Multi-context attention for human pose estimation. In *CVPR*, 2017.

[7] Haoshu Fang, Shuqin Xie, Yu-Wing Tai, and Cewu Lu. Rmpe: Regional multi-person pose estimation. In *ICCV*, 2017.

[8] Ross Girshick. Fast r-cnn. In *ICCV*, 2015.

[9] Ross Girshick, Ilija Radosavovic, Georgia Gkioxari, Piotr Dollár, and Kaiming He. Detectron. `https://github.com/facebookresearch/detectron`, 2018.

[10] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.

[11] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[13] Shaoli Huang, Mingming Gong, and Dacheng Tao. A coarse-fine network for keypoint localization. In *ICCV*, 2017.

[14] Eldar Insafutdinov, Leonid Pishchulin, Bjoern Andres, Mykhaylo Andriluka, and Bernt Schiele. Deepercut: A deeper, stronger, and faster multi-person pose estimation model. In *ECCV*, 2016.

[15] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *ACMMM*, 2014.

[16] Muhammed Kocabas, Salih Karagoz, and Emre Akbas. Multiposenet: Fast multi-person pose estimation using pose residual network. In *ECCV*, 2018.

[17] Tsung-Yi Lin, Piotr Dollár, Ross B Girshick, Kaiming He, Bharath Hariharan, and Serge J Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017.

[18] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.

[19] Gyeongsik Moon, Juyong Chang, and Kyoung Mu Lee. Posefix: Model-agnostic general human pose refinement network. In *CVPR*, 2019.

[20] Alejandro Newell, Zhiao Huang, and Jia Deng. Associative embedding: End-to-end learning for joint detection and grouping. In *NIPS*, 2017.

[21] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *ECCV*, 2016.

[22] George Papandreou, Tyler Zhu, Nori Kanazawa, Alexander Toshev, Jonathan Tompson, Chris Bregler, and Kevin Murphy. Towards accurate multi-person pose estimation in the wild. In *CVPR*, 2017.

[23] Chao Peng, Tete Xiao, Zeming Li, Yuning Jiang, Xiangyu Zhang, Kai Jia, Gang Yu, and Jian Sun. Megdet: A large mini-batch object detector. In *CVPR*, 2018.

[24] Leonid Pishchulin, Eldar Insafutdinov, Siyu Tang, Bjoern Andres, Mykhaylo Andriluka, Peter V Gehler, and Bernt Schiele. Deepcut: Joint subset partition and labeling for multi person pose estimation. In *CVPR*, 2016.

[25] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015.

[26] Matteo Ruggero Ronchi and Pietro Perona. Benchmarking and error diagnosis in multi-instance pose estimation. In *ICCV*, 2017.

[27] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

[28] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 2015.

[29] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *AAAI*, 2017.

[30] Jonathan J Tompson, Arjun Jain, Yann LeCun, and Christoph Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. In *NIPS*, 2014.

[31] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. In *CVPR*, 2014.

[32] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *CVPR*, 2016.

[33] Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking. *ECCV*, 2018.

[34] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *CVPR*, 2017.