Joint COCO and Mapillary Workshop at ICCV 2019: COCO Keypoint Detection Challenge Track

Technical Report: HRNet++

Naoki Kato Hideki Okada Yusuke Uchida
DeNA Co., Ltd.
{naoki.kato, hideki.okada, yusuke.a.uchida}@dena.com

Abstract

In this technical report, we present our solution for the COCO Keypoint Detection Task in the Joint COCO and Mapillary Recognition Challenge 2019. We extend HRNet in multiple ways to enhance the performance of the network. In addition, we apply soft-argmax operation to the output heatmap of the model to train human keypoints in an end-to-end manner, which enables precise localization of the keypoints. Our final model achieves 77.0 AP on the COCO validation set.

1. Introduction

Multi-person pose estimation aims at detecting and localizing keypoints for all persons in an image, which is useful for many applications. This task is challenging because it requires accurate localization of the keypoints of an unknown number of persons in situations where there may be a variety of lighting, clothing, human poses and occlusions.

Along with the progress of convolutional neural networks [5], the performance of pose estimation algorithms has also greatly improved. In particular, the recent top-down approaches which have large receptive fields over respective person instances show significant performance improvements over the bottom-up approaches [4, 20]. However, there is still room for improvement of the model performance. The bottleneck for the state-of-the-art methods lies in the localization error, especially small errors around the correct keypoint positions [12]. Recent mainstream heatmap-based methods tend to suffer from such errors since a keypoint position is predicted by taking a maximum position from a heatmap downsampled in a network. Although removing downsample layers in a model seems to cope with this problem, such a network requires higher computational costs. To deal with such a problem and improve localization accuracy, we employ a soft-argmax operation on the head of a network to predict the positions of the keypoints, similar to Sun et al [18]. Since the predicted keypoint location is continuous, this approach does not suffer from the quantization problem. We also modify the architecture of HRNet [16] in various ways to improve the model performance. We call the resulting model HRNet++. We demonstrate the effectiveness of our approach in our experiments.

2. Method

We adopt a two stage top-down approach to estimate multi-person keypoints. At the first stage, we detect human bounding boxes with a person detector. We then perform keypoint detection on the detected boxes at the second stage. In the following section, we describe the details of the person detector and the keypoint detector.

2.1. Person detector

We adopt HRNetV2p-W48 [17] with Hybrid Task Cascade architecture [2] to detect person boxes. This detector is trained using instance annotations and COCO-stuff [1] annotations in the COCO dataset. The detector achieves the detection AP of 59.6 for the person category on the COCO validation set.

2.2. Keypoint detector

We use HRNet as our baseline model and make some changes in its architecture and keypoint prediction method to improve the model performance. Illustration of our keypoint detector is depicted in Figure 1.

2.2.1 Network architecture

We extend HRNet architecture in multiple ways. We call our updated version of the network HRNet++. We describe each reformation in the following.

Introduce SE residual unit. Hu et al. demonstrated the effectiveness of SE block that adaptively recalibrates channel-wise feature responses in convolutional neural networks [6].

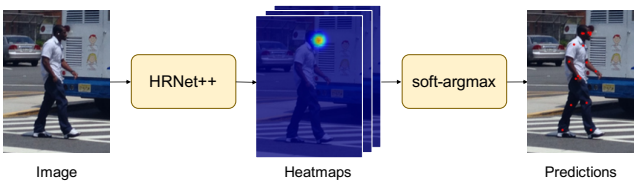

Figure 1: Our keypoint detector.

Based on that, we replace all the residual units in HRNet with SE residual units. The reduction ratio in the SE residual unit is set as $r = 8$.

Modify downsampling method. Original HRNet takes strided convolution with 3×3 filter to downsample feature maps. However, upsampling after the strided convolution with an odd convolutional filter will move the entire feature map to the lower right, which is problematic when adding the upsampled shifted map with the unshifted map [3]. To avoid this misalignment problem, we simply replace all the strided convolutions except the first two convolutional layers in HRNet with average pooling followed by 1×1 convolution.

Skip-connection across modules. To train a deeper and more accurate model, connections between layers close to the input and those close to the output are proven to be effective in vision tasks [13, 10]. HRNet, which extracts features from multi-resolution feature maps with parallel subnetworks, is suitable for taking such a structure. Hence, we introduce skip-connection across modules like DenseNet [7]. Specifically, outputs of an exchange block are used as the inputs of all subsequent exchange blocks that have the same feature resolution. Skipped features are fused to the original inputs by summation after 1×1 convolution that has the same number of output channels as the original inputs.

Modify head architecture. Original HRNet estimates keypoint heatmaps by the last exchange unit followed by 1×1 convolution. However, since the exchange unit fuses multi-resolution feature maps by summation, it is concerned that this architecture produces artifacts from low-resolution feature maps in resulting heatmaps. We cope with this problem by making modifications to the head architecture of HRNet, similar to DNANet [9]. To this end, we use sub-pixel convolution [15] to upsample low-resolution features maps and concatenate them with high-resolution features of adjacent subnetworks.

2.2.2 Keypoint prediction

We also modify the keypoint prediction method as well as the network architecture. To this end, we directly regress keypoint positions by applying soft-argmax (i.e. taking the center of mass) operation on the normalized heatmaps. Normalization is done by applying softmax with inverse temperature parameter α, which determines the sensitivity of the predicted position to the heatmap value. We set $\alpha = 60$. The model is trained using L1 loss on the predicted keypoints. For better convergence, we first train the model with MSE over heatmaps before training regression. We describe the details of the training procedure in Section 3.

3. Experiments

Datasets. We use two publicly available datasets to train keypoint detectors.

The COCO dataset [11] provided by the challenge organizers consists of more than 200K images and 250K person instances labeled with 17 keypoints. Our models are trained on the train2017 set including 57K images with 150K person instances. We report evaluation results on the val2017 set including 5K images. The final submission score is evaluated on the test-challenge set including 20K images.

The AI Challenger dataset [19] is used as an external dataset. This dataset contains 300K images and 700K person instances labeled with 14 keypoints. The training set and validation set have 210K images and 30K images, respectively. We split the trainval set into 444K person instances for training and use the remaining 5K person instances for validation.

We only train our keypoint detector and use the pretrained person detector which is publicly available.

Evaluation metrics. We report evaluation results of AP, AP_{50}, AP_{75}, AP_M, AP_L and AR used as the challenge metrics of COCO dataset. The primary challenge metric AP is calculated from the mean AP over 10 Object Keypoint Similarity (OKS) thresholds, where OKS is calculated from the scale of the person and the distance between predicted keypoints and the ground-truth keypoints. AP_{50} and AP_{75} are AP at OKS 0.50 and 0.75, respectively. AP_M and AP_L represent AP for the medium and large size of persons, respectively. AR is the mean AR calculated similarly to AP.

Training. Data preprocessing and data augmentation approaches are almost the same as those taken by Sun et al [16]. We extend the human bounding box in height or width to a fixed aspect ratio: $height : width = 4 : 3$, and then crop the box from the image, which is resized to a fixed size of 384×288 as the model input. The data augmentation includes flipping, random rotation ($[-45°, 45°]$), random scale ($[0.65, 1.35]$) and half body data augmentation [21].

We use pretrained weights with ImageNet classification task [14]. Since we modify the architecture of HRNet, we only initialize first stage of HRNet++ with pretrained weights provided by Sun et al [16].

The Adam optimizer [8] is used. We first train a heatmap model on the AI Challenger dataset for 30 epochs with a learning rate 1e-3. We then train the model on the COCO

Table 1: Comparisons on the COCO validation set. "Pretrain" means pretrained layers with the ImageNet classification task. "TTA" indicates performing test time augmentation in the form of scaling and rotation.

Method	Pretrain	Extra data	Input size	AP	AP_{50}	AP_{75}	AP_M	AP_L	AR
Original HRNet-W48	wo/ head	N	384×288	76.3	90.8	82.9	72.3	83.4	81.2
HRNet++W48 (heatmap)	stage1	Y	384×288	76.5	**90.9**	**83.0**	72.7	83.5	81.4
HRNet++W48 (regression)	stage1	Y	384×288	76.7	**90.9**	82.9	72.8	**83.8**	81.5
HRNet++W48 (regression) + TTA	stage1	Y	384×288	76.9	90.6	82.8	73.2	**83.8**	81.7
ensemble	stage1	Y	384×288	76.7	**90.9**	**83.0**	72.8	**83.8**	81.5
ensemble + TTA	stage1	Y	384×288	**77.0**	90.7	**83.0**	**73.4**	**83.8**	**81.8**

train2017 dataset for 150 epochs with the initial learning rate 1e-3 and decay it to 5e-5 with cosine scheduling (denoted as heatmap model). Finally, we fine-tune our model with soft-argmax on the COCO dataset for 20 epochs changing learning rate from 2e-5 to 2e-7 with cosine scheduling (denoted as regression model).

Testing. As we described in Section 2, we first estimate person bounding boxes with the person detector. We then extend and crop the estimated boxes from the image and resize to a fixed size to feed it to our keypoint detector, same as the training phase.

When using the heatmap model, each keypoint location is predicted by adjusting the highest heatvalue location with a quarter offset in the direction from the highest response to the second highest response. We perform model ensemble by averaging predicted keypoint positions from the heatmap model and the regression model. We compute heatmaps by averaging the heatmaps of the original and flipped images for all the models. We perform additional test time augmentation in scaling ($\{0.8, 1.0, 1.2\}$) and rotation ($\{-10°, 10°\}$) by averaging coordinates to get the final predictions.

3.1. Results

We compare the performance of our approaches and the original HRNet on the COCO validation set. Experimental results are shown in Table 1. Our heatmap version of HRNet++ obtains AP of 76.5. Although the pretrained layers and the training datasets are different, it outperforms the original HRNet. We believe pretraining more layers of HRNet++ should introduce further performance gain. The regression version of HRNet++ outperforms the heatmap model by 0.2 points in AP, showing the effectiveness of soft-argmax operation on the output heatmaps. Our final ensemble model with test time augmentation in scaling and rotation achieves 77.0 AP.

4. Conclusion

In the COCO Keypoint Challenge 2019, we mainly focus on improving the localization accuracy of the model and adopting soft-argmax operation to predict human keypoints. Moreover, we modify the architecture of HRNet to improve the model performance. We demonstrated the effectiveness of our approach with the experiments. In the future, we will explore an effective way to train a model that directly estimates keypoint positions without heatmap pretraining. We will also study how to combine multiple datasets in training to further improve the model performance.

References

[1] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. COCO-Stuff: Thing and stuff classes in context. In *CVPR*, 2018. 1

[2] Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, et al. Hybrid task cascade for instance segmentation. In *CVPR*, 2019. 1

[3] Yunpeng Chen, Haoqi Fang, Bing Xu, Zhicheng Yan, Yannis Kalantidis, Marcus Rohrbach, Shuicheng Yan, and Jiashi Feng. Drop an octave: Reducing spatial redundancy in convolutional neural networks with octave convolution. *arXiv preprint arXiv:1904.05049*, 2019. 2

[4] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. *arXiv preprint arXiv:1711.07319*, 2017. 1

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1

[6] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, 2018. 1

[7] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *CVPR*, 2017. 2

[8] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 2

[9] Ping Yao Ge Chen Chuanguang Yang Huimin Li Li Fu Tianyao Zheng Kun Zhang, Peng He. Dnanet: Denormalized attention based multi-resolution network for human pose estimation. *arXiv preprint arXiv:1909.05090*, 2019. 2

[10] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 2

[11] 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. 2

[12] Matteo Ruggero Ronchi and Pietro Perona. Benchmarking and error diagnosis in multi-instance pose estimation. 2017. 1

[13] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, 2015. 2

[14] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *IJCV*, 2015. 2

[15] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *CVPR*, 2016. 2

[16] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *CVPR*, 2019. 1, 2

[17] Ke Sun, Yang Zhao, Borui Jiang, Tianheng Cheng, Bin Xiao, Dong Liu, Yadong Mu, Xinggang Wang, Wenyu Liu, and Jingdong Wang. High-resolution representations for labeling pixels and regions. *arXiv preprint arXiv:1904.04514*, 2019. 1

[18] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *ECCV*, 2018. 1

[19] Jiahong Wu, He Zheng, Bo Zhao, Yixin Li, Baoming Yan, Rui Liang, Wenjia Wang, Shipei Zhou, Guosen Lin, Yanwei Fu, et al. Ai challenger: a large-scale dataset for going deeper in image understanding. *arXiv preprint arXiv:1711.06475*, 2017. 2

[20] Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking. In *ECCV*, 2018. 1

[21] B. Yin Q. Peng T. Xiao Y. Du Z. Li X. Zhang G. Yu Z. Wang, W. Li and J. Sun. Mscoco keypoints challenge 2018. 2014. 2