Learning Local-Global Contextual Adaptation for Multi-Person Pose Estimation

Supplementary Material

Nan Xue[1,2] Tianfu Wu[2*] Gui-Song Xia[1,3] Liangpei Zhang[3]
[1] School of Computer Science, Wuhan University [2] Department of ECE, NC State University
[3] State Key Lab. of Information Engieering in Surveying, Mapping and Remote Sensing, Wuhan University
Code& Models available at: https://github.com/cherubicXN/logocap

1. Experimental Settings

We present the details of training and testing as follows.

Training. We train two LOGO-CAP networks with the ImageNet pretrained HRNet-W32 and HRNet-W48 [6] as the feature backbone respectively on the COCO-train-2017 dataset [4]. Common training specifications are used for simplicity in experiments. The Adam optimizer [3] is used with default coefficients $\beta_1 = 0.9$ and $\beta_2 = 0.999$. For both the backbones, the total number of epochs is set to 140 and the batch size is set to 12 images per GPU card. The same learning rate schedule is used for both models. The learning rate is initially set to 0.001 and then decayed to 10^{-4} and 10^{-5} at the 90-th and 120-th epoch respectively. We use 4 and 8 V100 GPUs to accelerate the training for the two LOGO-CAP models with HRNet-W32 and HRNet-W48 respectively. The resolution of training images is 512×512 and 640×640 for the two models respectively. Following the widely adopted experimental settings in [7], the data augmentations in training include (1) random rotation with the rotation degree from $-30°$ to $30°$, (2) random scaling with the factor in the range of $[0.75, 1.5]$, (3) random translation in the range $[-40\text{pix}, 40\text{pix}]$ along both x and y directions, and (4) random horizontal flipping with the probability of 0.5.

Similarly, for different hyperparamters such as the trade-off parameter λ in the total loss, we did not run computationally expensive hyperparameter optimization for simplicity.

Testing. We focus on the single-scale testing protocol in the COCO keypoint benchmark for the sake of efficient human pose estimation. In the testing phase, the short side of input images is resized to a specific length (*e.g.*, 384, 512, or 640 pixels) and keep unchanged the aspect ratio between the height and the width. As commonly adopted in many bottom-up pose estimation approaches (*e.g.*, AE [5], HrHRNet [1], DEKR [2]), the flip testing is used as our default setting for the fair comparison. In the implementation, we feed the stacked tensor with an input image and a horizontally-flipped one together to get the global heatmaps and the offset fields. The flipped outputs are then averaged (according to the flip index) to get the final global heatmaps and the offset fields. For the computation of local heatmaps and the local-global adaptation, only the non-flipped outputs are used for the final predictions.

*Corresponding author

Table 1. The performance of a vanilla center-offset regression approach, its empirical upper bound, and the performance of our proposed LOGO-CAP using HRNet-W32 [6] as the feature backbone. See text for detail.

	Baseline	Emp. Bound	LOGO-CAP
AP	60.1	88.9	70.0
AP^{50}	85.2	93.1	88.2
AP^{75}	66.7	90.6	76.4
AP^{M}	53.7	87.7	64.4
AP^{L}	71.5	90.2	78.4

2. The Empirical Upper Bound

We elaborate on the details of computing the empirical upper bound of performance for a vanilla center-offset pose estimation method. The detailed results are reported in Tab. 1.

Network Architecture. The vanilla center-offset regression baseline uses the ImageNet pretrained HRNet-W32 [6] as the backbone, and the same modules as in our LOGO-CAP+HRNet-W32 for the center heatmap regression and the offset vector regression. We present the details of computing keypoint expansion maps (KEMs) that are used in calculating the empirical uppper bound as follows.

Computation of Keypoint Expansion Maps. Denoted by $\mathcal{P} \in \mathbb{R}^{N\times 17\times 2}$ the initial pose parameters (i.e., the 2-D locations for the 17 keypoints of the N pose instances)

estimated by the vanilla center-offset method, we expand each of the estimated keypoints with a local 11×11 mesh grid, that is to lift a keypoint to a 2-D mesh to counter the estimation uncertainty. We use the COCO benchmark provided keypoint sigmas to scale the unit length of the meshgrid for different types (*e.g.*, nose, eyes, hips) of keypoints. After getting the expanded keypoint meshes $\mathcal{M}$ of the initial poses, we compute their keypoint similarities $\mathcal{S} \in \mathbb{R}^{N\times 11\times 11\times K\times 17}$ between the groundtruth keypoints $\mathcal{G} \in \mathbb{R}^{K\times 17\times 3}$ and the keypoint expansion maps. By applying the sum reduction on the similarity tensor $\mathcal{S}$ along the 2-nd, 3-rd and the last axes, we have known the optimal correspondence (including the low-quality matches) for each center anchor, denoted by $\mathcal{S}_{N\times 11\times 11\times 17}$. Then, the pose with the maximal similarity in the 11×11 local window for each center anchor are used as the best one to compute the empirical upper bound on the fully-annotated COCO-val-2017 dataset.

3. Multi-Scale Testing Results

We report the multi-scale testing results following the protocol used in DEKR, as well as the number of parameters and the GFLOPs in Tab. 2. Overall, the multi-scale testing scheme improves the results on the COCO benchmark. For the OCHuman dataset, it is shown that the multi-scale testing scheme downgrades the performance for both our W48 model and the two DEKR models. It should be noted that the multi-scale testing for bottom-up approaches will lead to a very slow inference speed (even slower than the top-down approaches) for both DEKR and our method. Although the AP scores can be improved on COCO, it is out of the pursuit of better speed-accuracy trade-off for bottom-up paradigms.

Table 2. The results of the single-scale (s.s.) and multi-scale (m.s.) testing, the number of parameters and GFLOPs.

	COCO-val-2017		COCO-testdev-2017		OCHuman-val		OCHuman-test	
	AP (s.s.)	AP (m.s.)	AP (s.s.)	AP (m.s.)	AP (s.s.)	AP (m.s.)	AP (s.s.)	AP (m.s.)
Ours (W32)	69.6	71.3	68.2	69.9	39.0	40.6	38.1	39.9
Ours (W48)	**72.2**	**73.2**	**70.8**	**71.9**	**41.2**	**40.9**	**40.4**	**40.1**
DEKR (W32)	68.0	70.7	67.3	69.8	37.9	36.6	36.5	36.2
DEKR (W48)	71.0	72.3	70.0	71.0	38.8	37.0	38.2	36.3

4. More Qualitative Results

Fig. 1 shows some qualitative examples of human pose estimation by the proposed LOGO-CAP on the COCO-val-2017 dataset. For each image in Fig. 1, we select a person instance to show the OKS difference between the initial pose and the refined pose in a close-up visualization when matching to the ground truth poses.

Results on the COCO-val-2017 and the OCHuman Datasets. Fig. 2 shows examples of pose estimation in the two datasets by the proposed LOGO-CAP with the HRNet-W32 backbone. Our proposed LOCO-CAP is able to handle large structural and appearance variations in human pose estimation.

Fast pose estimation for video frames. To justify the potential of our proposed approach in practical applications, we run our LOGO-CAP (W32 model) on two videos that have the resolution of 1280×720 from YouTube. We follow our testing protocol to resize the short side of the video frames to 512 pixels and keep their original aspect ratios for inference. Without using any pose tracking techniques, our LOGO-CAP achieves fast and accurate human pose estimation. Please click the following anonymous links for the demo videos (**with background musics**):

- https://bit.ly/3cFcJ75 (video credit: https://youtu.be/2DiQUX11YaY)
- https://bit.ly/30RkyEg (video credit: https://youtu.be/kTvzU1sGSyA)

In these two demo videos, the instantaneous FPS for each video frame is marked in the left corner of the video.

References

[1] Bowen Cheng, Bin Xiao, Jingdong Wang, Honghui Shi, Thomas S. Huang, and Lei Zhang. Higherhrnet: Scale-aware representation learning for bottom-up human pose estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5385–5394. IEEE, 2020. 1

[2] Zigang Geng, Ke Sun, Bin Xiao, Zhaoxiang Zhang, and Jingdong Wang. Bottom-up human pose estimation via disentangled keypoint regression. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 1

[3] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *International Conference on Learning Representations (ICLR)*, 2015. 1

[4] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In David J. Fleet, Tomás Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *European Conference on Computer Vision (ECCV)*, volume 8693, pages 740–755, 2014. 1

[5] Alejandro Newell, Zhiao Huang, and Jia Deng. Associative embedding: End-to-end learning for joint detection and grouping. In *Advances in Neural Information Processing Systems 30 (NeurIPS)*, pages 2277–2287, 2017. 1

[6] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5693–5703, 2019. 1

[7] Feng Zhang, Xiatian Zhu, Hanbin Dai, Mao Ye, and Ce Zhu. Distribution-aware coordinate representation for human pose estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7091–7100, 2020. 1

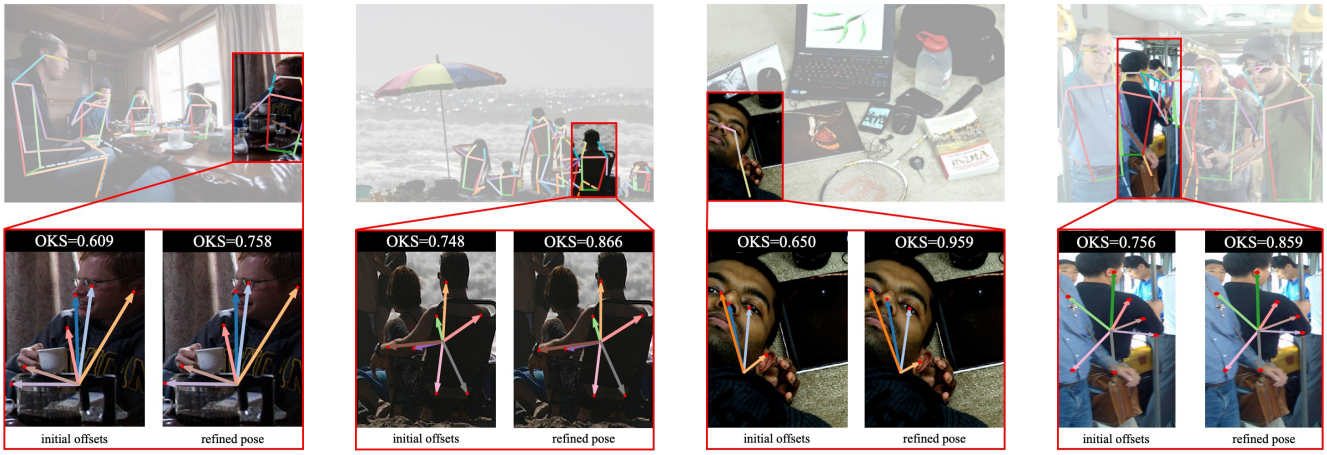

Figure 1. Examples of human pose estimation in the COCO val-2017 dataset by the proposed LOGO-CAP with the HRNet-W32 backbone. *Top:* The COCO skeleton template based visualization. *Bottom*: The close-up visualization and OKS comparisons between the initial center-offset estimation and the refined keypoints.

Figure 2. Qualitative results of our LOGO-CAP (HRNet-W32). All images were picked thematically without considering our algorithms performance. The first two rows display our approach on the COCO-val-2017 dataset and the last two ones show our results on the OCHuman test dataset.