Cascaded Pyramid Network for Multi-Person Pose Estimation

Yilun Chen* Zhicheng Wang* Yuxiang Peng[1] Zhiqiang Zhang[2] Gang Yu Jian Sun
[1]Tsinghua University [2]HuaZhong University of Science and Technology
Megvii Inc. (Face++), {chenyilun, wangzhicheng, pyx, zhangzhiqiang, yugang, sunjian}@megvii.com

Abstract

The topic of multi-person pose estimation has been largely improved recently, especially with the development of convolutional neural network. However, there still exist a lot of challenging cases, such as occluded keypoints, invisible keypoints and complex background, which cannot be well addressed. In this paper, we present a novel network structure called Cascaded Pyramid Network (CPN) which targets to relieve the problem from these "hard" keypoints. More specifically, our algorithm includes two stages: GlobalNet and RefineNet. GlobalNet is a feature pyramid network which can successfully localize the "simple" keypoints like eyes and hands but may fail to precisely recognize the occluded or invisible keypoints. Our RefineNet tries explicitly handling the "hard" keypoints by integrating all levels of feature representations from the GlobalNet together with an online hard keypoint mining loss. In general, to address the multi-person pose estimation problem, a top-down pipeline is adopted to first generate a set of human bounding boxes based on a detector, followed by our CPN for keypoint localization in each human bounding box. Based on the proposed algorithm, we achieve state-of-art results on the COCO keypoint benchmark, with average precision at 73.0 on the COCO test-dev dataset and 72.1 on the COCO test-challenge dataset, which is a 19% relative improvement compared with 60.5 from the COCO 2016 keypoint challenge. Code[1] and the detection results are publicly available for further research.

1. Introduction

Multi-person pose estimation is to recognize and locate the keypoints for all persons in the image, which is a fundamental research topic for many visual applications like human action recognition and human-computer interaction.

Recently, the problem of multi-person pose estimation has been greatly improved by the involvement of deep convolutional neural networks [22, 16]. For example, in [5], convolutional pose machine is utilized to locate the keypoint joints in the image and part affinity fields (PAFs) is proposed to assemble the joints to different person. Mask-RCNN [15] predicts human bounding boxes first and then warps the feature maps based on the human bounding boxes to obtain human keypoints. Although great progress has been made, there still exist a lot of challenging cases, such as occluded keypoints, invisible keypoints and crowded background, which cannot be well localized. The main reasons lie at two points: 1) these "hard" joints cannot be simply recognized based on their appearance features only, for example, the torso point; 2) these "hard" joints are not explicitly addressed during the training process.

To address these "hard" joints, in this paper, we propose a novel network structure called Cascaded Pyramid Network (CPN). There are two stages in our network architecture: GlobalNet and RefineNet. Our GlobalNet learns a good feature representation based on feature pyramid network [24]. More importantly, the pyramid feature representation can provide sufficient context information, which is inevitable for the inference of the occluded and invisible joints. Based on the pyramid features, our RefineNet explicitly address the "hard" joints based on an online hard keypoints mining loss.

Based on our Cascaded Pyramid Network, we address the multi-person pose estimation problem based on a top-down pipeline. Human detector is first adopted to generate a set of human bounding boxes, followed by our CPN for keypoint localization in each human bounding box. In addition, we also explore the effects of various factors which might affect the performance of multi-person pose estimation, including person detector and data preprocessing. These details are valuable for the further improvement of accuracy and robustness of our algorithm.

In summary, our contributions are three-fold as follows:

- We propose a novel and effective network called cascaded pyramid network (CPN), which integrates global pyramid network (GlobalNet) and pyramid refined network based on online hard keypoints min-

*:The first two authors contribute equally to this work. This work is done when Yilun Chen, Xiangyu Peng and Zhiqiang Zhang are interns at Megvii Research.

[1]https://github.com/chenyilun95/tf-cpn.git

ing (RefineNet)

- We explore the effects of various factors contributing to muti-person pose estimation involved in top-down pipeline.
- Our algorithm achieves state-of-art results in the challenging COCO multi-person keypoint benchmark, that is, 73.0 AP in test-dev dataset and 72.1 AP in test challenge dataset.

2. Related Work

Human pose estimation is an active research topic for decades. Classical approaches tackling the problem of human pose estimation mainly adopt the techniques of pictorial structures [10, 1] or graphical models [7]. More specifically, the classical works [1, 34, 13, 33, 8, 44, 29, 20] formulate the problem of human keypoints estimation as a tree-structured or graphical model problem and predict keypoint locations based on hand-crafted features. Recent works [27, 14, 4, 19, 39, 42] mostly rely on the development of convolutional neural network(CNN) [22, 16], which largely improve the performance of pose estimation. In this paper, we mainly focus on the methods based on the convolutional neural network. The topic is categorized as single-person pose estimation that predicts the human keypoints based on the cropped image given bounding box, and multi-person pose estimation that require further recognition of the full body poses of all persons in one image.

Multi-Person Pose Estimation. Multi-person pose estimation is gaining increasing popularity recently because of the high demand for the real-life applications. However, multi-person pose estimation is challenging owing to occlusion, various gestures of individual persons and unpredictable interactions between different persons. The approach of multi-person pose estimation is mainly divided into two categories: bottom-up approaches and top-down approaches.

Bottom-Up Approaches. Bottom-up approaches [5, 26, 30, 19] directly predict all keypoints at first and assemble them into full poses of all persons. DeepCut [30] interprets the problem of distinguishing different persons in an image as an Integer Linear Program (ILP) problem and partition part detection candidates into person clusters. Then the final pose estimation results are obtained when person clusters are combined with labeled body parts. DeeperCut [19] improves DeepCut [30] using deeper ResNet [16] and employs image-conditioned pairwise terms to get better performance. Zhe Cao *et al*. [5] map the relationship between keypoints into part affinity fields (PAFs) and assemble detected keypoints into different poses of people. Newell *et al*. [26] simultaneously produce score maps and pixel-wise embedding to group the candidate keypoints to different people to get final multi-person pose estimation.

Top-Down Approaches. Top-down approaches [28, 18, 15, 9] interpret the process of detecting keypoints as a two-stage pipeline, that is, firstly locate and crop all persons from image, and then solve the single person pose estimation problem in the cropped person patches. Papandreou *et al*. [28] predict both heatmaps and offsets of the points on the heatmaps to the ground truth location, and then uses the heatmaps with offsets to obtain the final predicted location of keypoints. Mask-RCNN [15] predicts human bounding boxes first and then crops the feature map of the corresponding human bounding box to predict human keypoints. If top-down approach is utilized for multi-person pose estimation, a human detector as well as single person pose estimator is important in order to obtain a good performance. Here we review some works about single person pose estimation and recent state-of-art detection methods.

Single Person Pose Estimation. Toshev *et al*. firstly introduce CNN to solve pose estimation problem in the work of DeepPose [38], which proposes a cascade of CNN pose regressors to deal with pose estimation. Tompson *et al*. [37] attempt to solve the problem by predicting heatmaps of keypoints using CNN and graphical models. Later works such as Wei *et al*. [40]and Newell *et al*. [27] show great performance via generating the score map of keypoints using very deep convolutional neural networks. Wei *et al*. [40] propose a multi-stage architecture, i.e., first generate coarse results, and continuously refine the result in the following stages. Newell *et al*. [27] propose an U-shape network, i.e., hourglass module, and stack up several hourglass modules to generate prediction. Carreira *et al*. [6] uses iterative error feedback to get pose estimation and refine the prediction gradually. Lifshitz *et al*. [23] uses deep consensus voting to vote the most probable location of keypoints. Gkioxary *et al*. [14] and Zisserman *et al*. [2] apply RNN-like architectures to sequentially refine the results. Our work is partly inspired by the works on generating and refining score maps. Yang *et al*. [43] adopts pyramid features as inputs of the network in the process of pose estimation, which is a good exploration of the utilization of pyramid features in pose estimation. However, more refinement operations are required to pyramid structure in pose estimation.

Human Detection. Human detection approaches are mainly guided by the RCNN family [12, 11, 31], the up-to-date detectors of which are [24, 15]. These detection approaches are composed of two-stage in general. First generate boxes proposals based on default anchors, and then crop from the feature map and further refine the proposals to get the final boxes via R-CNN network. The detector used in our methods are mostly based on [24, 15].

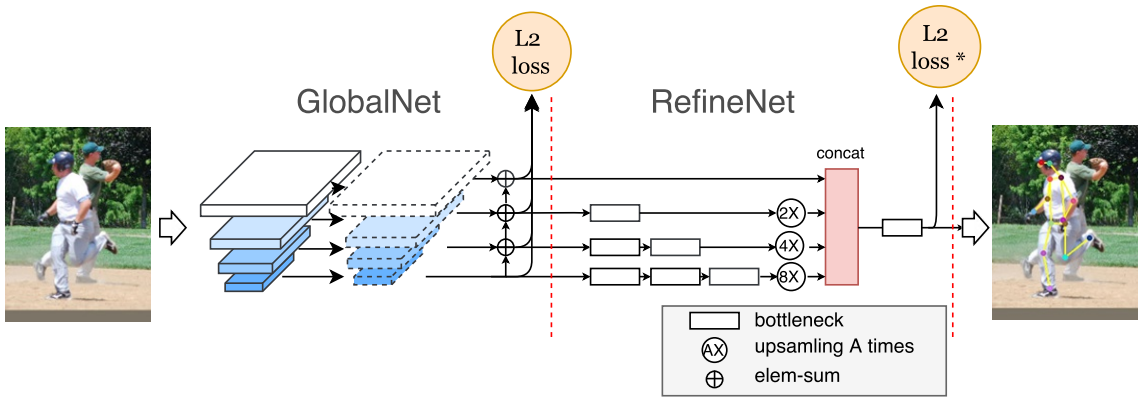

Figure 1. Cascaded Pyramid Network. "L2 loss*" means L2 loss with online hard keypoints mining.

3. Our Approach for Multi-perosn Keypoints Estimation

Similar to [15, 28], our algorithm adopts the top-down pipeline: a human detector is first applied on the image to generate a set of human bounding-boxes and detailed localization of the keypoints for each person can be predicted by a single-person skeleton estimator.

3.1. Human Detector

We adopt the state-of-art object detector algorithms based on FPN [24]. ROIAlign from Mask RCNN [15] is adopted to replace the ROIPooling in FPN. To train the object detector, all eighty categories from the COCO dataset are utilized during the training process but only the boxes of human category is used for our multi-person skeleton task. For fair comparison with our algorithms, we will release the detector results on the COCO val and COCO test dataset.

3.2. Cascaded Pyramid Network (CPN)

Before starting the discussion of our CPN, we first briefly review the design structure for the single person pose estimator based on each human bounding box. Stacked hourglass [27], which is a prevalent method for pose estimation, stacks eight hourglasses which are down-sampled and up-sampled modules with residual connections to enhance the pose estimation performance. The stacking strategy works to some extent, however, we find that stacking two hourglasses is sufficient to have a comparable performance compared with the eight-stage stacked hourglass module. [28] utilizes a ResNet [16] network to estimate pose in the wild achieving promising performance in the COCO 2016 keypoint challenge. Motivated by the works [27, 28] described above, we propose an effective and efficient network called cascaded pyramid network (CPN) to address the problem of pose estimation. As shown in Figure 1, our CPN involves two sub-networks: GlobalNet and RefineNet.

3.2.1 GlobalNet

Here, we describe our network structure based on the ResNet backbone. We denote the last residual blocks of different conv features conv2∼5 as $C_2, C_3, ..., C_5$ respectively. 3×3 convolution filters are applied on $C_2, ..., C_5$ to generate the heatmaps for keypoints. As shown in Figure 2, the shallow features like C_2 and C_3 have the high spatial resolution for localization but low semantic information for recognition. On the other hand, deep feature layers like C_4 and C_5 have more semantic information but low spatial resolution due to strided convolution (and pooling). Thus, usually an U-shape structure is integrated to maintain both the spatial resolution and semantic information for the feature layers. More recently, FPN [24] further improves the U-shape structure with deeply supervised information. We apply the similar feature pyramid structure for our keypoints estimation. Slightly different from FPN, we apply 1×1 convolutional kernel before each element-wise sum procedure in the upsampling process. We call this structure as GlobalNet and an illustrative example can be found in Figure 1.

As shown in Figure 2, our GlobalNet based on ResNet backbone can effectively locate the keypoints like eyes but may fail to precisely locate the position of hips. The localization of keypoints like hip usually requires more context information and processing rather than the nearby appearance feature. There exists many cases that are difficult to directly recognize these "hard" keypoints by a single GlobalNet.

3.2.2 RefineNet

Based on the feature pyramid representation generated by GlobalNet, we attach a RefineNet to explicitly address the "hard" keypoints. In order to improve the efficiency and keep integrity of information transmission, our RefineNet transmits the information across different levels and finally

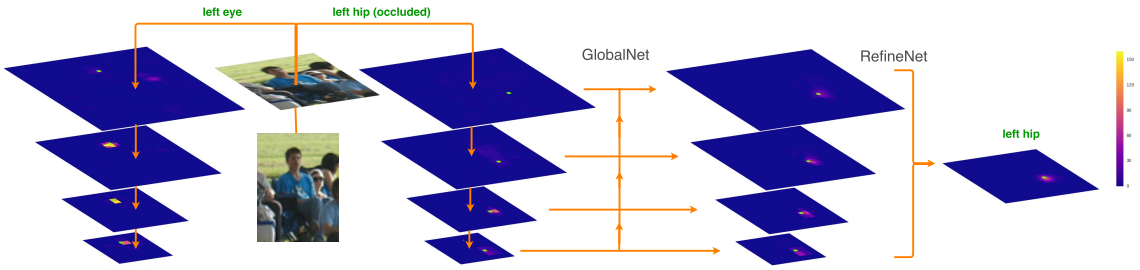

Figure 2. Output heatmaps from different features. The green dots means the groundtruth location of keypoints.

integrates the informations of different levels via upsampling and concatenating as HyperNet [21]. Different from the refinement strategy like stacked hourglass [27], our RefineNet concatenates all the pyramid features rather than simply using the upsampled features at the end of hourglass module. In addition, we stack more bottleneck blocks into deeper layers, whose smaller spatial size achieves a good trade-off between effectiveness and efficiency.

As the network continues training, the network tends to pay more attention to the "simple" keypoints of the majority but less importance to the occluded and hard keypoints. We should ensure the network balance between these two type of keypoints. Thus, in our RefineNet, we explictily select the hard keypoints online based on the training loss (which we called online hard keypoints mining) and back-propagate the gradients from the selected keypoints only.

4. Experiment

Our overall pipeline follows the top-down approach for estimating multiple human poses. Firstly, we apply a state-of-art bounding detector to generate human proposals. For each proposal, we assume that there is only one main person in the cropped region of proposal and then applied the pose estimating network to generate the final prediction. In this section, we will discuss more details of our methods based on experiment results.

4.1. Experimental Setup

Dataset and Evaluation Metric. Our models are only trained on MS COCO[25] trainval dataset (includes 57K images and 150K person instances) and validated on MS COCO minival dataset (includes 5000 images). The testing sets includes test-dev set (20K images) and test-challenge set (20K images). Most experiments are evaluated in OKS-based mAP, where OKS (object keypoints similarity) defines the similarity between different human poses.

Cropping Strategy. For each human detection box, the box is extended to a fixed aspect ratio, e.g., height : width = 256 : 192, and then we crop from images without distorting the images aspect ratio. Finally, we resize the cropped image to a fixed size of height 256 pixels and 192 pixels by default. Note that only the boxes of the person class in the top 100 boxes of all classes are used in all the experiments of 4.2.

Data Augmentation Strategy. Data augmentation is critical for the learning of scale invariance and rotation invariance. After cropping from images, we apply random flip, random rotation ($-45^\circ \sim +45^\circ$) and random scale ($0.7 \sim 1.35$).

Training Details. All models of pose estimation are trained using adam algorithm with an initial learning rate of 5e-4. Note that we also decrease the learning rate by a factor of 2 every 3600000 iteration. We use a weight decay of 1e-5 and the training batch size is 32. Batch normalization is used in our network. Generally, the training of ResNet-50-based models takes about 1.5 day on eight NVIDIA Titan X Pascal GPUs. Our models are all initialized with weights of the public-released ImageNet [32]-pretrained model.

Testing Details. In order to minimize the variance of prediction, we apply a gaussian filter on the predicted heatmaps. Following the same techniques used in [27], we also predict the pose of the corresponding flipped image and average the heatmaps to get the final prediction; a quarter offset in the direction from the highest response to the second highest response is used to obtain the final location of the keypoints. Rescoring strategy is also used in our experiments. Different from the rescoring strategy used in [28], the product of boxes' score and the average score of all keypoints is considered as the final pose score of a person instance.

4.2. Ablation Experiment

In this subsection, we'll validate the effectiveness of our network from various aspects. Unless otherwise specified, all experiments are evaluated on MS COCO minival dataset in this subsection. The input size of all models is 256×192 and the same data augmentation is adopted.

4.2.1 Person Detector

Since detection boxes are critical for top-down approaches in multi-person pose estimation, here we discuss two factors of detection, i.e. different NMS strategies and the AP

of bounding boxes. Our human boxes are generated based on the state-of-art detector FPN trained with only the labeled COCO data, no extra data and no specific training on person. For fair comparison, we use the same detector with a general AP of 41.1 and person AP of 55.3 on the COCO minival dataset in the ablation experiments by default unless otherwise specified.

Non-Maximum Suppression (NMS) strategies. As shown in the Table 1, we compare the performance of different NMS strategies or the same NMS strategy under different thresholds. Referring to the original hard NMS, the performance of keypoints detection improves when the threshold increases, basically owing to the improvement of the average precision (AP) and average recall (AR) of the boxes. Since the final score of the pose estimated partially depends on the score of the bounding box, Soft-NMS [3] which is supposed to generate more proper scores is better in performance as it is shown in the Table 1 . From the table, we can see that Soft-NMS [3] surpasses the hard NMS method on the performance of both detection and keypoints detection.

NMS	AP(all)	AP(H)	AR(H)	AP(OKS)
NMS(thr=0.3)	40.1	53.5	60.3	68.2
NMS(thr=0.4)	40.5	54.4	61.7	68.9
NMS(thr=0.5)	40.8	54.9	62.9	69.2
NMS(thr=0.6)	40.8	55.2	64.3	69.2
Soft-NMS [3]	41.1	55.3	67.0	69.4

Table 1. Comparison between different NMS methods and keypoints detection performance with the same model. H is short for human.

Detection Performance. Table 2 shows the relationship between detection AP and the corresponding keypoints AP, aiming to reveal the influence of the accuracy of the bounding box detection on the keyoints detection. From the table, we can see that the keypoints detection AP gains less and less as the accuracy of the detection boxes increases. Specially, when the detection AP increases from 44.3 to 49.3 and the human detection AP increases 3.0 points, the keypoints detection accuracy does not improve a bit and the AR of the detection increases marginally. Therefore, we have enough reasons to deem that the given boxes cover most of the medium and large person instances with such a high detection AP. Therefore, the more important problem for pose estimation is to enhance the accuracy of hard keypoints other than involve more boxes.

4.2.2 Cascaded Pyramid Network

8-stage hourglass network [27] and ResNet-50 with dilation [28] are adopted as our baseline. From Table 3, although the results improve considerably if dilation are used in shallow layers, it is worth noting that the FLOPs (floating-point operations) increases significantly.

Det Methods	AP(all)	AP(H)	AR(H)	AP(OKS)
FPN-1	36.3	49.6	58.5	68.8
FPN-2	41.1	55.3	67.0	69.4
FPN-3	44.3	58.4	71.3	69.7
ensemble-1	49.3	61.4	71.8	69.8
ensemble-2	52.1	62.9	74.7	69.8

Table 2. Comparison between detection performance and keypoints detection performance. FPN-1: FPN with the backbone of Res50; FPN-2: Res101 with Soft-NMS and OHEM [35] applied; FPN-3: ResNeXt [41]101 with Soft-NMS, OHEM [35], multiscale training applied; ensemble-1: multiscale test involved; ensemble-2: multiscale test, large batch and SENet [17] involved. H is short for Human.

Models	AP (OKS)	FLOPs	Param Size
1-stage hourglass	54.5	3.92G	12MB
2-stage hourglass	66.5	6.14G	23MB
8-stage hourglass	66.9	19.48G	89MB
ResNet-50	41.3	3.54G	92MB
ResNet-50 + dilation(res5)	44.1	5.62G	92MB
ResNet-50 + dilation(res4-5)	66.5	17.71G	92MB
ResNet-50 + dilation(res3-5)	–	68.70G	92MB
GlobalNet only (ResNet-50)	66.6	3.90G	94MB
CPN* (ResNet-50)	68.6	6.20G	102 MB
CPN (ResNet-50)	69.4	6.20G	102 MB

Table 3. Results on COCO minival dataset. CPN* indicates CPN without online hard keypoints mining.

From the statistics of FLOPs in testing stage and the accuracy of keypoints as shown in Table 3, we find that CPN achieves much better speed-accuracy trade-off than Hourglass network and ResNet-50 with dilation. Note that GlobalNet achieves much better results than one-stage hourglass network of same FLOPs probably for much larger parameter space. After refined by the RefineNet, it increases 2.0 AP and yields the results of 68.6 AP. Furthermore, when online hard keypoints mining is applied in RefineNet, our network finally achieves 69.4 AP.

Design Choices of RefineNet. Here, we compare different design strategies of RefineNet as shown in Table 4. We compare the following implementation based on pyramid output from the GlobalNet:

1) Concatenate (Concat) operation is directly attached like HyperNet [21],

2) Only one bottleneck block is attached first in each layer ($C_2 \sim C_5$) and then followed by a concatenate operation,

3) Different number of bottleneck blocks applied to dif-

ferent layers followed by a concatenate operation as shown in Figure 1.

A convolution layer is attached finally to generate the score maps for each keypoint.

We find that our RefineNet structure can effectively achieve more than 2 points gain compared with GlobalNet only and for refinement of keypoints and also outperforms other design implementations followed by GlobalNet.

Models	AP(OKS)	FLOPs
GlobalNet only	66.6	3.90G
GlobalNet + Concat	68.5	5.87G
GlobalNet + 1 bottleneck +Concat	69.2	6.92G
ours (CPN)	69.4	6.20G

Table 4. Comparison of models of different design choices of RefineNet.

Here, we also validate the performance for utilizing the pyramid output from different levels. In our RefineNet, we utilize four output feature maps $C_2 \sim C_5$, where C_i refers to the ith feature map of GlobalNet output. Also, feature map from C_2 only, feature maps from $C_2 \sim C_3$, and feature maps from $C_2 \sim C_4$ are evaluated as shown in Table 5. We can find that the performance improves as more levels of features are utilized.

Connections	AP(OKS)	FLOPs
C_2	68.3	5.02G
$C_2 \sim C_3$	68.4	5.50G
$C_2 \sim C_4$	69.1	5.88G
$C_2 \sim C_5$	69.4	6.20G

Table 5. Effectiveness of intermediate connections between GlobalNet and RefineNet.

4.2.3 Online Hard Keypoints Mining

Here we discuss the losses used in our network. In detail, the loss function of GlobalNet is L2 loss of all annotated keypoints while the second stage tries learning the hard keypoints, that is, we only punish the top $M(M < N)$ keypoint losses out of N (the number of annotated keypoints in one person, say 17 in COCO dataset). The effect of M is shown in Table 6. For $M = 8$, the performance of second stage achieves the best result for the balanced training between hard keypoints and simple keypoints.

M	6	8	10	12	14	17
AP (OKS)	68.8	69.4	69.0	69.0	69.0	68.6

Table 6. Comparison of different hard keypoints number in online hard keypoints mining.

Inspired by OHEM [35], however the method of online hard keypoints mining loss is essentially different from it. Our method focuses on higher level information than OHEM which concentrates on examples, for instance, pixel level losses in the heatmap L2 loss. As a result, our method is more stable, and outperforms OHEM strategy in accuracy.

As Table 7 shows, when online hard keypoints mining is applied in RefineNet, the performance of overall network increases 0.8 AP and finally achieves 69.4 AP comparing to normal l2 loss. For reference, experiments without intermediate supervision in CPN leads to a performance drop of 0.9 AP probably for the lack of prior knowledge and sufficient context information of keypoints provided by GlobalNet. In addition, applying the same online hard keypoints mining in GlobalNet which decreases the results by 0.3 AP.

GlobalNet	RefineNet	AP(OKS)
—	L2 loss	68.2
L2 loss	L2 loss	68.6
—	L2 loss*	68.5
L2 loss	L2 loss*	69.4
L2 loss*	L2 loss*	69.1

Table 7. Comparison of models with different losses function. Here "-" denotes that the model applies no loss function in corresponding subnetwork. "L2 loss*" means L2 loss with online hard keypoints mining.

4.2.4 Data Pre-processing

The size of cropped image are important factors to the performance of keypoints detection. As Table 8 illustrates, it's worth noting that the input size 256×192 actually works as well as 256×256 which costs more computations of almost 2G FLOPs using the same cropping strategy. As the input size of the cropped images increases, more location details of human keypoints are fed into the network resulting in a large performance improvement. Additionally, online hard keypoints mining works better when the input size of the crop images is enlarged by improving 1 point on 384×288 input size.

Models	Input Size	FLOPs	AP(OKS)
8-stage Hourglass	256×192	19.5G	66.9
8-stage Hourglass	256×256	25.9G	67.1
CPN* (ResNet-50)	256×192	6.2G	68.6
CPN (ResNet-50)	256×192	6.2G	69.4
CPN* (ResNet-50)	384×288	13.9G	70.6
CPN (ResNet-50)	384×288	13.9G	71.6

Table 8. Comparison of models of different input size. CPN* indicates CPN without online hard keypoints mining.

Methods	AP	$AP_{@.5}$	$AP_{@.75}$	AP_m	AP_l	AR	$AR_{@.5}$	$AR_{@.75}$	AR_m	AR_l
FAIR Mask R-CNN*	68.9	89.2	75.2	63.7	76.8	75.4	93.2	81.2	70.2	82.6
G-RMI*	69.1	85.9	75.2	66.0	74.5	75.1	90.7	80.7	69.7	82.4
bangbangren+*	70.6	88.0	76.5	65.6	**79.2**	77.4	93.6	83.0	71.8	**85.0**
oks*	71.4	89.4	78.1	65.9	79.1	77.2	93.6	83.4	71.8	84.5
Ours+ (CPN+)	**72.1**	**90.5**	**78.9**	**67.9**	78.1	**78.7**	**94.7**	**84.8**	**74.3**	84.7

Table 9. Comparisons of final results on COCO test-challenge2017 dataset. "*" means that the method involves extra data for training. Specifically, FAIR Mask R-CNN involves distilling unlabeled data, oks uses AI-Challenger keypoints dataset, bangbangren and G-RMI use their internal data as extra data to enhance performance. "+" indicates results using ensembled models. The human detector of Ours+ is a detector that has an AP of 62.9 of human class on COCO minival dataset. CPN and CPN+ in this table all use the backbone of ResNet-Inception [36] framework.

Methods	AP	$AP_{@.5}$	$AP_{@.75}$	AP_m	AP_l	AR	$AR_{@.5}$	$AR_{@.75}$	AR_m	AR_l
CMU-Pose [5]	61.8	84.9	67.5	57.1	68.2	66.5	87.2	71.8	60.6	74.6
Mask-RCNN [15]	63.1	87.3	68.7	57.8	71.4	-	-	-	-	-
Associative Embedding [26]	65.5	86.8	72.3	60.6	72.6	70.2	89.5	76.0	64.6	78.1
G-RMI [28]	64.9	85.5	71.3	62.3	70.0	69.7	88.7	75.5	64.4	77.1
G-RMI* [28]	68.5	87.1	75.5	65.8	73.3	73.3	90.1	79.5	68.1	80.4
Ours (CPN)	72.1	91.4	80.0	68.7	77.2	78.5	**95.1**	85.3	74.2	84.3
Ours+ (CPN+)	**73.0**	**91.7**	**80.9**	**69.5**	**78.1**	**79.0**	**95.1**	**85.9**	**74.8**	**84.7**

Table 10. Comparisons of final results on COCO test-dev dataset. "*" means that the method involves extra data for training. "+" indicates results using ensembled models. The human detectors of Our and Ours+ the same detector that has an AP of 62.9 of human class on COCO minival dataset.CPN and CPN+ in this table all use the backbone of ResNet-Inception [36] framework.

Methods	AP - minival	AP - dev	AP - challenge
Ours (CPN)	72.7	72.1	-
Ours (CPN+)	74.5	73.0	72.1

Table 11. Comparison of results on the minvival dataset and the corresponding results on test-dev or test-challenge of the COCO dataset. "+" indicates ensembled model. CPN and CPN+ in this table all use the backbone of ResNet-Inception [36] framework.

4.3. Results on MS COCO Keypoints Challenge

We evaluate our method on MS COCO test-dev and test-challenge dataset. Table 10 illustrates the results of our method in the test-dev split dataset of the COCO dataset. Without extra data involved in training, we achieve 72.1 AP using a single model of CPN and 73.0 using ensembled models of CPN with different ground truth heatmaps. Table 9 shows the comparison of the results of our method and the other methods on the test-challenge2017 split of COCO dataset. We get 72.1 AP achieving state-of-art performance on COCO test-challenge2017 dataset. Table 11 shows the performances of CPN and CPN+ (ensembled model) on COCO minival dataset, which offer a reference to the gap between the COCO minival dataset and the standard test-dev or test-challenge dataset of the COCO dataset. Figure 3 illustrates some results generated using our method.

5. Conclusion

In this paper, we follow the top-down pipeline and a novel Cascaded Pyramid Network (CPN) is presented to address the "hard" keypoints. More specifically, our CPN includes a GlobalNet based on the feature pyramid structure and a RefineNet which concatenates all the pyramid features as a context information. In addition, online hard keypoint mining is integrated in RefineNet to explicitly address the "hard" keypoints. Our algorithm achieves state-of-art results on the COCO keypoint benchmark, with average precision at 73.0 on the COCO test-dev dataset and 72.1 on the COCO test-challenge dataset, outperforms the COCO 2016 keypoint challenge winner by a 19% relative improvement.

References

[1] M. Andriluka, S. Roth, and B. Schiele. Pictorial structures revisited: People detection and articulated pose estimation. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1014–1021, 2009. 2

[2] V. Belagiannis and A. Zisserman. Recurrent human pose estimation. pages 468–475, 2016. 2

[3] N. Bodla, B. Singh, R. Chellappa, and L. S. Davis. Improving Object Detection With One Line of Code. *ArXiv e-prints*, Apr. 2017. 5

[4] A. Bulat and G. Tzimiropoulos. *Human Pose Estimation via Convolutional Part Heatmap Regression*. Springer International Publishing, 2016. 2

Figure 3. Some results of our method.

[5] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, 2017. 1, 2, 7

[6] J. Carreira, P. Agrawal, K. Fragkiadaki, and J. Malik. Human pose estimation with iterative error feedback. 2013(2013):4733–4742, 2015. 2

[7] X. Chen and A. Yuille. Articulated pose estimation by a graphical model with image dependent pairwise relations. *Eprint Arxiv*, pages 1736–1744, 2014. 2

[8] M. Dantone, J. Gall, C. Leistner, and L. Van Gool. Human pose estimation using body parts dependent joint regressors. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2013. 2

[9] H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu. Rmpe: Regional multi-person pose estimation. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 2

[10] M. A. Fischler and R. A. Elschlager. The representation and matching of pictorial structures. *IEEE Transactions on Computers*, C-22(1):67–92, 2006. 2

[11] R. Girshick. Fast R-CNN. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2015. 2

[12] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 2

[13] G. Gkioxari, P. Arbelaez, L. Bourdev, and J. Malik. Articulated pose estimation using discriminative armlet classifiers. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3342–3349, 2013. 2

[14] G. Gkioxari, A. Toshev, and N. Jaitly. Chained predictions using convolutional neural networks. In *European Conference on Computer Vision*, pages 728–743, 2016. 2

[15] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask R-CNN. *arXiv preprint arXiv:1703.06870*, 2017. 1, 2, 3, 7

[16] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 1, 2, 3

[17] J. Hu, L. Shen, and G. Sun. Squeeze-and-Excitation Networks. 5

[18] S. Huang, M. Gong, and D. Tao. A coarse-fine network for keypoint localization. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 2

[19] E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, and B. Schiele. Deepercut: A deeper, stronger, and faster multi-person pose estimation model. In *European Conference on Computer Vision*, pages 34–50, 2016. 2

[20] S. Johnson and M. Everingham. Learning effective human pose estimation from inaccurate annotation. In *Computer Vision and Pattern Recognition*, pages 1465–1472, 2011. 2

[21] T. Kong, A. Yao, Y. Chen, and F. Sun. Hypernet: Towards accurate region proposal generation and joint object detection. In *Computer Vision and Pattern Recognition*, pages 845–853, 2016. 4, 5

[22] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 1, 2

[23] I. Lifshitz, E. Fetaya, and S. Ullman. Human pose estimation using deep consensus voting. In *European Conference on Computer Vision*, pages 246–260, 2016. 2

[24] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 1, 2, 3

[25] T. Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollr, and C. L. Zitnick. Microsoft coco: Common objects in context. 8693:740–755, 2014. 4

[26] A. Newell, Z. Huang, and J. Deng. Associative Embedding: End-to-End Learning for Joint Detection and Grouping. *ArXiv e-prints*, Nov. 2016. 2, 7

[27] A. Newell, K. Yang, and J. Deng. Stacked hourglass networks for human pose estimation. In *European Conference on Computer Vision*, pages 483–499, 2016. 2, 3, 4, 5

[28] G. Papandreou, T. Zhu, N. Kanazawa, A. Toshev, J. Tompson, C. Bregler, and K. Murphy. Towards Accurate Multi-person Pose Estimation in the Wild. *ArXiv e-prints*, Jan. 2017. 2, 3, 4, 5, 7

[29] L. Pishchulin, M. Andriluka, P. Gehler, and B. Schiele. Poselet conditioned pictorial structures. In *Computer Vision and Pattern Recognition*, pages 588–595, 2013. 2

[30] L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. Gehler, and B. Schiele. Deepcut: Joint subset partition and labeling for multi person pose estimation. In *Computer Vision and Pattern Recognition*, pages 4929–4937, 2016. 2

[31] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Neural Information Processing Systems (NIPS)*, 2015. 2

[32] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. 4

[33] B. Sapp, C. Jordan, and B. Taskar. Adaptive pose priors for pictorial structures. In *Computer Vision and Pattern Recognition*, pages 422–429, 2010. 2

[34] B. Sapp and B. Taskar. Modec: Multimodal decomposable models for human pose estimation. In *Computer Vision and Pattern Recognition*, pages 3674–3681, 2013. 2

[35] A. Shrivastava, A. Gupta, and R. Girshick. Training region-based object detectors with online hard example mining. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 761–769, 2016. 5, 6

[36] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. *ArXiv e-prints*, Feb. 2016. 7

[37] J. Tompson, A. Jain, Y. Lecun, and C. Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. *Eprint Arxiv*, pages 1799–1807, 2014. 2

[38] A. Toshev and C. Szegedy. Deeppose: Human pose estimation via deep neural networks. pages 1653–1660, 2013. 2

[39] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. Convolutional pose machines. In *CVPR*, 2016. 2

[40] S. E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. Convolutional pose machines. pages 4724–4732, 2016. 2

[41] S. Xie, R. Girshick, P. Dollar, Z. Tu, and K. He. Aggregated residual transformations for deep neural networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 5

[42] W. Yang, S. Li, W. Ouyang, H. Li, and X. Wang. Learning feature pyramids for human pose estimation. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 2

[43] W. Yang, S. Li, W. Ouyang, H. Li, and X. Wang. Learning feature pyramids for human pose estimation. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 2

[44] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *Computer Vision and Pattern Recognition*, pages 1385–1392, 2011. 2