MultiPoseNet: Fast Multi-Person Pose Estimation using Pose Residual Network

Muhammed Kocabas[0000−0001−8593−0415], Salih Karagoz[0000−0002−7438−8322], and Emre Akbas[0000−0002−3760−6722]

Department of Computer Engineering, Middle East Technical University
{muhammed.kocabas,e234299,eakbas}@metu.edu.tr

Abstract. In this paper, we present *MultiPoseNet*, a novel bottom-up multi-person pose estimation architecture that combines a multi-task model with a novel assignment method. MultiPoseNet can jointly handle person detection, person segmentation and pose estimation problems. The novel assignment method is implemented by the *Pose Residual Network (PRN)* which receives keypoint and person detections, and produces accurate poses by assigning keypoints to person instances. On the COCO keypoints dataset, our pose estimation method outperforms all previous bottom-up methods both in accuracy (+4-point mAP over previous best result) and speed; it also performs on par with the best top-down methods while being at least 4x faster. Our method is the fastest real time system with ∼23 frames/sec.

Keywords: Multi-Task Learning · Multi-Person Pose Estimation · Semantic Segmentation · MultiPoseNet · Pose Residual Network

1 Introduction

This work is aimed at estimating the two-dimensional (2D) poses of multiple people in a given image. Any solution to this problem has to tackle a few subproblems: (i) detecting body joints (or keypoints, as they are called in the widely used COCO [36] dataset) such as wrists, ankles, etc., (ii) grouping these joints into person instances, or detecting people and (iii) assigning joints to person instances. Depending on which sub-problem is tackled first, there have been two major approaches in multi-person 2D estimation: *bottom-up* and *top-down*. Bottom-up methods [5,6,25,26,37,39,42] first detect body joints without having any knowledge as to the number of people or their locations. Next, detected joints are grouped to form individual poses for person instances. On the other hand, top-down methods [10,18,23,40] start by detecting people first and then for each person detection, a single-person pose estimation method (e.g. [12,24,38,48]) is executed. Single-person pose estimation, i.e. detecting body joints conditioned on the information that there is a single person in the given input (the top-down approach), is typically a more costly process than grouping the detected joints (the bottom-up approach). Consequently, the top-down methods tend to

be slower than the bottom-up methods, since they need to repeat the single-person pose estimation for each person detection; however, they usually yield better accuracy than bottom-up methods.
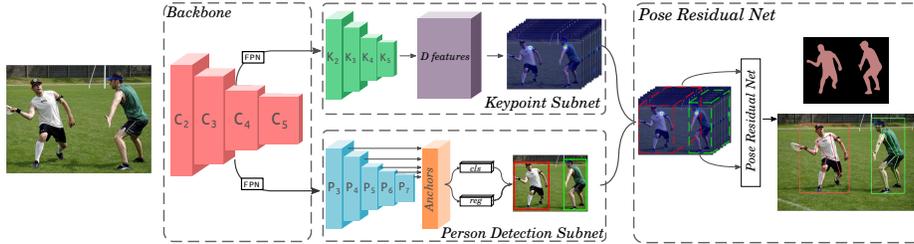

Fig. 1. MultiPoseNet is a multi-task learning architecture capable of performing human keypoint estimation, detection and semantic segmentation tasks altogether efficiently.

In this paper, we present a new bottom-up method (with respect to the categorization given above) for multi-person 2D pose estimation. Our method is based on a multi-task learning model, which can jointly handle the person detection, person segmentation and pose estimation problems. To emphasize its multi-person and multi-task aspects of our model, we named it as "MultiPoseNet. Our model (Fig. 1) consists of a shared backbone for feature extraction, detection subnets for keypoint and person detection/segmentation, and a final network which carries out the pose estimation, i.e. assigning detected keypoints to person instances. Our major contribution lies in the pose estimation step where the network implements a novel assignment method. This network receives keypoint and person detections, and produces a pose for each detected person by assigning keypoints to person boxes using a learned function. In order to put our contribution into context, here we briefly describe the relevant aspects of the state-of-the-art (SOTA) bottom-up methods [6,37]. These methods attempt to group detected keypoints by exploiting lower order relations either between the group and keypoints, or among the keypoints themselves. Specifically, Cao et al. [6] model pairwise relations (called part affinity fields) between two nearby joints and the grouping is achieved by propagating these pairwise affinities. In the other SOTA method, Newell et al. [37] predict a real number called a *tag* per detected keypoint, in order to identify the group the detection belongs to. Hence, this model makes use of the unary relations between a certain keypoint and the group it belongs to. Our method generalizes these two approaches in the sense that we achieve the grouping in a single shot by considering all joints together at the same time. We name this part of our model which achieves the grouping as the *Pose Residual Network* (PRN) (Fig. 2). PRN takes a region-of-interest (RoI) pooled keypoint detections and then feeds them into a residual multilayer perceptron (MLP). PRN considers all joints simultaneously and learns configurations of joints. We illustrate this capability of PRN by plotting a sample

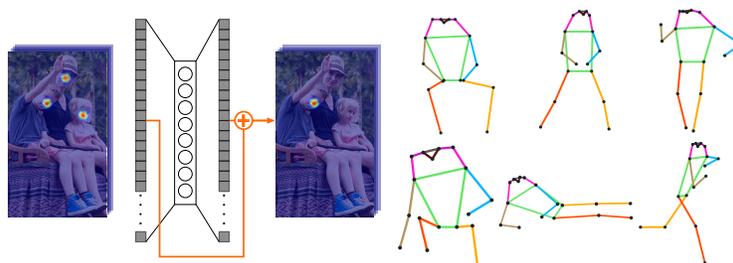set of learned configurations. (Fig. 2 right). Our experiments (on the COCO

Fig. 2. Left: Pose Residual Network (PRN). The PRN is able to disambiguate which keypoint should be assigned to the current person box. **Right:** Six sample poses obtained via clustering the structures learned by PRN.

dataset, using no external data) show that our method outperforms all previous bottom-up methods: we achieve a 4-point mAP increase over the previous best result. Our method performs on par with the best performing top-down methods while being at least 4x faster than them. To the best of our knowledge, there are only two top-down methods that we could not outperform. Given the fact that bottom-up methods have always performed less accurately than the top-down methods, our results are remarkable. In terms of running time, our method appears to be the fastest of all multi-person 2D pose estimation methods. Depending on the number of people in the input image, our method runs at between 27 frames/sec (FPS) (for one person detection) and 15 FPS (for 20 person detections). For a typical COCO image, which contains ∼3 people on average, we achieve ∼23 FPS (Fig. 6). Our contributions in this work are four fold. (1) We propose the *Pose Residual Network* (PRN), a simple yet very effective method for the problem of assigning/grouping body joints. (2) We outperform all previous bottom-up methods and achieve comparable performance with top-down methods. (3) Our method works faster than all previous methods, in real-time at ∼23 frames/sec. (4) Our network architecture is extendible; we show that using the same backbone, one can solve other related problems, too, e.g. person segmentation.

2   Related Work

2.1   Single Person Pose Estimation

Single person pose estimation is to predict individual body parts given a cropped person image (or, equivalently, given its exact location and scale within an image). Early methods (prior to deep learning) used hand-crafted HOG features [14] to detect body parts and probabilistic graphical models to represent the pose structure (tree-based [2,28,41,51]; non-tree based [15,21]). Deep neural networks

based models [3, 7, 13, 24, 33, 38, 45, 46, 48, 51] have quickly dominated the pose estimation problem after the initial work by Toshev et al. [46] who used the AlexNet architecture to directly regress spatial joint coordinates. Tompson et al. [45] learned pose structure by combining deep features along with graphical models. Carreira et al. [7] proposed the Iterative Error Feedback method to train Convolutional Neural Networks (CNNs) where the input is repeatedly fed to the network along with current predictions in order to refine the predictions. Wei et al. [48] were inspired by the pose machines [43] and used CNNs as feature extractors in pose machines. *Hourglass blocks*, (HG) developed by Newell et al. [38], are basically convolution-deconvolution structures with residual connections. Newell et al. stacked HG blocks to obtain an iterative refinement process and showed its effectiveness on single person pose estimation. Stacked Hourglass (SHG) based methods made a remarkable performance increase over previous results. Chu et al. [13] proposed adding visual attention units to focus on keypoint regions of interest. Pyramid residual modules by Yang et al. [51] improved the SHG architecture to handle scale variations. Lifshitz et al. [33] used a probabilistic keypoint voting scheme from image locations to obtain agreement maps for each body part. Belagiannis et al. [3] introduced a simple recurrent neural network based prediction refinement architecture. Huang et al. [24] developed a coarse-to-fine model with Inception-v2 [44] network as the backbone. The authors calculated the loss in each level of the network to learn coarser to finer representations of parts.

2.2   Multi Person Pose Estimation

Bottom-up Multi person pose estimation solutions branched out as bottom-up and top-down methods. Bottom-up approaches detect body joints and assign them to people instances, therefore they are faster in test time and smaller in size compared to top-down approaches. However, they miss the opportunity to zoom into the details of each person instance. This creates an accuracy gap between top-down and bottom-up approaches. In an earlier work by Ladicky et al. [32], they proposed an algorithm to jointly predict human part segmentations and part locations using HOG-based features and probabilistic approach. Gkioxari et al. [20] proposed k-poselets to jointly detect people and keypoints. Most of the recent approaches use Convolutional Neural Networks (CNNs) to detect body parts and relationships between them in an end-to-end manner [6,25,37,41,42,47], then use assignment algorithms [6, 25, 42, 47] to form individual skeletons. Pischulin et al. [42] used deep features for joint prediction of part locations and relations between them, then performed correlation clustering. Even though [42] doesn't use person detections, it is very slow due to proposed clustering algorithm and processing time is in the order of hours. In a follow-up work by Insafutdinov et al. [25], they benefit from deeper ResNet architectures as part detectors and improved the parsing efficiency of a previous approach with an incremental optimization strategy. Different from Pischulin and Insafutdinov, Iqbal et al. [27] proposed to solve the densely connected graphical model locally, thus improved

time efficiency significantly. Cao et al. [6] built a model that contains two entangled CPM [48] branches to predict keypoint heatmaps and pairwise relationships (part affinity fields) between them. Keypoints are grouped together with fast Hungarian bipartite matching algorithm according to conformity of part affinity fields between them. This model runs in realtime. Newell et al. [37] extended their SHG idea by outputting associative vector embeddings which can be thought as tags representing each keypoint's group. They group keypoints with similar tags into individual people.

Top-down Top-down methods first detect people (typically using a top performing, off-the-shelf object detector) and then run a single person pose estimation (SPPEN) method per person to get the final pose predictions. Since a SPPEN model is run for each person instance, top-down methods are extremely slow, however, each pose estimator can focus on an instance and perform fine localization. Papandreou et al. [40] used ResNet with dilated convolutions [22] which has been very successful in semantic segmentation [8] and computing keypoint heatmap and offset outputs. In contrast to Gaussian heatmaps, the authors estimated a disk-shaped keypoint masks and 2-D offset vector fields to accurately localize keypoints. Joint part segmentation and keypoint detection given human detections approach were proposed by Xia et al. [49] The authors used separate PoseFCN and PartFCN to obtain both part masks and locations and fused them with fully-connected CRFs. This provides more consistent predictions by eliminating irrelevant detections. Fang et al. [18] proposed to use spatial transformer networks to handle inaccurate bounding boxes and used *stacked hourglass* blocks [38]. He et al. [23] combined instance segmentation and keypoint prediction in their *Mask-RCNN* model. They append keypoint heads on top of *RoI aligned* feature maps to get a one-hot mask for each keypoint. Chen et al. [10] developed *globalnet* on top of *Feature Pyramid Networks* [34] for multiscale inference and refined the predictions by using hyper-features [31].

3   The Method and Models

The architecture of our proposel model, MultiPoseNet, can be found in Fig. 1. In the following, we describe each component in detail.

3.1   The Shared Backbone

The backbone of MultiPoseNet serves as a feature extractor for keypoint and person detection subnets. It is actually a ResNet [22] with two Feature Pyramid Networks (FPN) [34] (one for the keypoint subnet, the other for the person detection subnet) connected to it, FPN creates pyramidal feature maps with top-down connections from all levels of CNNs feature hierarchy to make use of inherent multi-scale representations of a CNN feature extractor. By doing so, FPN compromises high resolution, weak representations with low resolution, strong representations. Powerful localization and classification properties of FPN

proved to be very successful in detection, segmentation and keypoint tasks recently [10, 23, 34, 35]. In our model, we extracted features from the last residual blocks C_2, C_3, C_4, C_5 with strides of (4,8,16,32) pixels and compute corresponding FPN features per subnet.

3.2   Keypoint Estimation Subnet

Keypoint estimation subnet (Fig. 3) takes hierarchical CNN features (outputted by the corresponding FPN) and outputs keypoint and segmentation heatmaps. Heatmaps represent keypoint locations as Gaussian peaks. Each heatmap layer belongs to a specific keypoint class (nose, wrists, ankles etc.) and contains arbitrary number of peaks that pertain to person instances. Person segmentation mask at the last layer of heatmaps encodes the pixelwise spatial layout of people in the image. A set of features specific to the keypoint detection task are com-
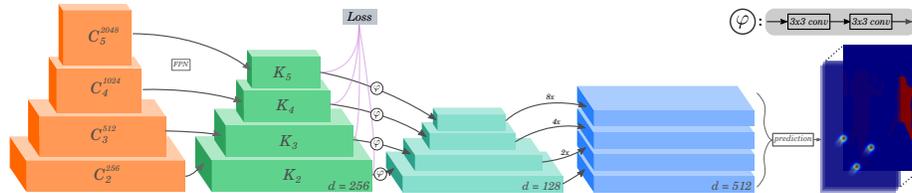

Fig. 3. The architecture of the keypoint subnet. It takes hierarchical CNN features as input and outputs keypoint and segmentation heatmaps.

puted similarly to [34] with top-down and lateral connections from the bottom-up pathway. $K_2 - K_5$ features have the same spatial size corresponding to $C_2 - C_5$ blocks but the depth is reduced to 256. K features are identical to P features in the original FPN paper, but we denote them with K to distinguish from person detection subnet layers. The depth of P features is downsized to 128 with 2 subsequent 3×3 convolutions to obtain D_2, D_3, D_4, D_5 layers. Since D features still have different strides, we upsampled D_3, D_4, D_5 accordingly to match 4-pixel stride as D_2 features and concatenated them into a single depth-512 feature map. Concatenated features are smoothed by a 3×3 convolution with ReLU. Final heatmap which has $(K + 1)$ layers obtained via 1×1 convolutions without activation. The final output is multiplied with a binary mask of $\mathbf{W}$ which has $\mathbf{W}(\mathbf{p}) = 0$ in the area of the persons without annotation. K is the number of human keypoints annotated in a dataset and $+1$ is person segmentation mask. In addition to the loss applied in the last layer, we append a loss at each level of K features to benefit from intermediate supervision. Semantic person segmentation masks are predicted in the same way with keypoints.

3.3 Person Detection Subnet

In order to design a faster and simpler person detection model which is compatible with FPN backbone, we have adopted RetinaNet. Same strategies to compute anchors, losses and pyramidal image features are followed. Classification and regression heads are modified to handle only person annotations.

<div align="center">(a)   (b)   (c)   (d)</div>

Fig. 4. Bounding box overlapping scenarios.

3.4 Pose Residual Network (PRN)

Assigning keypoint detections to person instances (bounding boxes, in our case) is straightforward if there is only one person in the bounding box as in Fig. 4 a-b. However, it becomes non-trivial if there are overlapping people in a single box as in Fig. 4 c-d. In the case of an overlap, a bounding box can contain multiple keypoints not related to the person in question, and this creates ambiguity in constructing final pose predictions. We solve these ambiguities by learning pose structures from data. The input to PRN is prepared as follows. For each person box that the person detection subnet detected, the region from the keypoint detection subnet's output, corresponding to the box, is cropped and resized to a fixed size, which ensures that PRN can handle person detections of arbitrary sizes and shapes. Specifically, let $\mathbf{X}$ denote the input to the PRN, where $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k\}$ in which $\mathbf{x}_k \in \mathbb{R}^{W \times H}$, k is the number of different keypoint types. The final goal of PRN is to output $\mathbf{Y}$ where $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_k\}$, in which $\mathbf{y}_k \in \mathbb{R}^{W \times H}$ is of the same size as $\mathbf{x}_k$, containing the correct position for each keypoint indicated by a peak in that keypoints channel. PRN models the mapping from $\mathbf{X}$ to $\mathbf{Y}$ as

$$\mathbf{y}_k = \phi_k(\mathbf{X}) + \mathbf{x}_k \tag{1}$$

where the functions $\phi_1(\cdot), \ldots, \phi_K(\cdot)$ apply a *residual correction* to the pose in $\mathbf{X}$, hence the name pose residual network. We implement Eq. 1 using a residual multilayer perceptron (Fig. 2). Activation of the output layer uses softmax to obtain a proper probability distribution and binary cross-entropy loss is used during training. Before we came up with this residual model, we experimented with two naive baselines and a non-residual model. In the first baseline method, which we call *Max*, for each keypoint channel k, we find the location with the

highest value and place a Gaussian in the corresponding location of the kth channel in $\mathbf{Y}$. In the second baseline method, we compute $\mathbf{Y}$ as

$$\mathbf{y}_k = \mathbf{x}_k * \mathbf{P}_k \tag{2}$$

where $\mathbf{P}_k$ is a prior map for the location of the kth joint, learned from ground-truth data and $*$ is element-wise multiplication. We named this method as Unary Conditional Relationship (UCR). Finally, in our non-residual model, we implemented

$$\mathbf{y}_k = \phi_k(\mathbf{X}). \tag{3}$$

Performances of all these models can be found in Table 3. In the context of the models described above, both SOTA bottom up methods learn lower order grouping models than the PRN. Cao et al. [6] model pairwise channels in $\mathbf{X}$ while Newell et al. [37] model only unary channels in $\mathbf{X}$. Hence, our model can be considered as a generalization of these lower order grouping models. We hypothesize that each node in PRN's hidden layer encodes a certain body configuration. To show this, we visualized some of the representative outputs of PRN in Fig. 2. These poses are obtained via reshaping PRN outputs and selecting the maximum activated keypoints to form skeletons. All obtained configurations are clustered using k-means with OKS (object keypoint similarity) [36] and cluster means are visualized in Fig. 2. OKS (object keypoint similarity) is used as k-means distance metric to cluster the meaningful poses.

3.5   Implementation Details

Training Due to different convergence times and loss imbalance, we have trained keypoint and person detection tasks separately. To use the same backbone in both task, we first trained the model with only keypoint subnet Fig. 3. Thereafter, we froze the backbone parameters and trained the person detection subnet. Since the two tasks are semantically similar, person detection results were not adversely affected by the frozen backbone. We used the Tensorflow [1] and Keras [11] deep learning libraries. For person detection, we made use of open-source Keras RetinaNet [19] implementation.

Keypoint Estimation Subnet: For keypoint training, we used 480x480 image patches, that are centered around the crowd or the main person in the scene. Random rotations between ± 40 degrees, random scaling between $0.8 - 1.2$ and vertical flipping with a probability of 0.3 were used during training. We have transferred the ImageNet [16] pretrained weights for each backbone before training. We optimize the model with Adam [30] starting with a learning rate of 1e-4 and decreased it by a factor of 0.1 in plateaux. We used the Gaussian peaks located at the keypoint locations as the ground truth to calculate L_2 loss, and we masked (ignored) people that are not annotated. We appended the segmentation masks to ground-truth as an extra layer and trained along with keypoint heatmaps. The cost function that we minimize is

$$L_{kp} = \mathbf{W} \cdot \|\mathbf{H}_t - \mathbf{H}_p\|_2^2, \tag{4}$$

where $\mathbf{H}_t$ and $\mathbf{H}_p$ are the ground-truth and predicted heatmaps respectively, and $\mathbf{W}$ is the mask used to ignore non-annotated person instances.

Person Detection Subnet: We followed a similar person detection training strategy as Lin et al. [35]. Images containing persons are used, they are resized such that shorter edge is 800 pixels. We froze backbone weights after keypoint training and not updated them during person detection training. We optimized subnet with Adam [30] starting with a learning rate of 1e-5 and is decreased by a factor of 0.1 in plateaux. We used Focal loss with ($\gamma = 2, \alpha = 0.25$) and smooth L_1 loss for classification and bbox regression, respectively. We obtained final proposals using NMS with a threshold of 0.3.

Pose Residual Network: During training, we cropped input and output pairs and resized heatmaps according to bounding-box proposals. All crops are resized to a fixed size of 36×56 (height/width = 1.56). We trained the PRN network separately and Adam optimizer [30] with a learning rate of 1e-4 is used during training. Since the model is shallow, convergence takes 1.5 hours approximately. We trained the model with person instances having at least 2 keypoints. We utilized a sort of curriculum learning [4] by sorting annotations based on number of keypoints and bounding box areas. In each epoch, model is started to learn easy-to-predict instances, hard examples are given in later stages.

Inference The whole architecture (see in Fig. 1) behaves as a monolithic, end-to-end model during test time. First, an image ($W \times H \times 3$) is processed through backbone model to extract the features in multi-scales. Person and keypoint detection subnets compute outputs simultaneously out of extracted features. Keypoints are outputted as $W \times H \times (K+1)$ sized heatmaps. K is the number of keypoint channels, and $+1$ is for the segmentation channel. Person detections are in the form of $N \times 5$, where N is the number of people and 5 channel corresponds to 4 bounding box coordinates along with confidence scores. Keypoint heatmaps are cropped and resized to form RoIs according to person detections. Optimal RoI size is determined as $36 \times 56 \times K$ in our experiments. PRN takes each RoI as separate input, then outputs same size RoI with only one keypoint selected in each layer of heatmap. All selected keypoints are grouped as a person instance.

4   Experiments

4.1   Datasets

We trained our keypoint and person detection models on COCO keypoints dataset [36] (without using any external/extra data) in our experiments. We used COCO for evaluating the keypoint and person detection, however, we used PASCAL VOC 2012 [17] for evaluating person segmentation due to the lack of semantic segmentation annotations in COCO. Backbone models (ResNet-50

and ResNet-101) were pretrained on ImageNet and we finetuned with COCO-keypoints. COCO train2017 split contains 64K images including 260K person instances which 150K of them having keypoint annotations. Keypoints of persons with small area are not annotated in COCO. We did ablation experiments on COCO val2017 split which contains 2693 images with person instances. We made comparison to previous methods on the test-dev2017 split which has 20K test images. We evaluated test-dev2017 results on the online COCO evaluation server. We use the official COCO evaluation metric average precision (AP) and average recall (AR). OKS and IoU based scores were used for keypoint and person detection tasks, respectively. We performed person segmentation evaluation in PASCAL VOC 2012 test split with PASCAL IoU metric. PASCAL VOC 2012 person segmentation test split contains 1456 images. We obtained test results using the online evaluation server.

4.2   Multi Person Pose Estimation

The overall AP results of our method along with top-performing bottom-up (BU) and top-down (TD) methods are given in Table 1. MultiPoseNet outperforms all bottom-up methods and most of the top-down methods. We outperform the previously best bottom-up method [37] by a 4-point increase in mAP. In addition, the runtime speed (see the FPS column Table 1 and Fig. 6) of our system is far better than previous methods with 23 FPS on average[1]. This proves the effectiveness of PRN for assignment and our multitask detection approach while providing reasonable speed-accuracy tradeoff. To get these results (Table 1) on test-dev, we have utilized test time augmentation and ensembling (as also done in all previous studies). Multi scale and multi crop testing was performed during test time data augmentation. Two different backbones and a single person pose refinement network similar to our keypoint detection model was used for ensembling. Results from different models are gathered and redundant detections was removed via OKS based NMS [40]. With ablation experiments we have inspected the effect of different backbones, keypoint detection architectures, and PRN designs. Tables 2 and 3 present the ablation analysis results on the COCO validation set. We present the recall-precision curves of our method for different scales *all, large, medium* in the supplementary material.

Different Backbones We used ResNet models [22] as shared backbone to extract features. Table 2 shows the impact of deeper features and dilated features. R101 improves the result by 1.6 mAP over R50. Dilated convolutions [8] which are very successful in dense detection tasks increase accuracy by 2 mAP over the R50 architecture. However, dilated convolutional filters add more computational complexity, consequently hinder realtime performance. We showed that concatenation of K features and intermediate supervision (Section 3.2) is crucial

---

[1] We obtained the FPS results by averaging the inference time using images containing 3 people on a 1080Ti GPU. We got CFNs and Mask RCNNs FPS results from their respective papers.

Table 1. Results on COCO **test-dev**, excluding systems trained with external data. Top-down methods are shown separately to make a clear comparison between bottom-up methods.

		FPS	AP	AP$_{50}$	AP$_{75}$	AP$_M$	AP$_L$	AR	AR$_{50}$	AR$_{75}$	AR$_M$	AR$_L$
BU	Ours	23	**69.6**	86.3	**76.6**	**65.0**	**76.3**	**73.5**	88.1	**79.5**	**68.6**	**80.3**
BU	Newell et al. [37]	6	65.5	**86.8**	72.3	60.6	72.6	70.2	**89.5**	76.0	64.6	78.1
BU	CMU-Pose [6]	10	61.8	84.9	67.5	57.1	68.2	66.5	87.2	71.8	60.6	74.6
TD	Megvii [10]	-	73.0	91.7	80.9	69.5	78.1	79.0	95.1	85.9	74.8	84.6
TD	CFN [24]	3	72.6	86.7	69.7	78.3	64.1	-	-	-	-	-
TD	Mask R-CNN [23]	5	69.2	90.4	76.0	64.9	76.3	75.2	93.7	81.1	70.3	81.8
TD	SJTU [18]	0.4	68.8	87.5	75.9	64.6	75.1	73.6	91.0	79.8	68.9	80.2
TD	GRMI-2017 [40]	-	66.9	86.4	73.6	64.0	72.0	71.6	89.2	77.6	66.1	79.1
TD	G-RMI-2016 [40]	-	60.5	82.2	66.2	57.6	66.6	66.2	86.6	71.4	61.9	72.2

for good perfomance. The results demonstrate that performance of our system can be further enhanced with stronger feature extractors like recent ResNext [50] architectures.

Different Keypoint Architectures

Keypoint estimation requires dense prediction over spatial locations, so its performance is dependent on input and output resolution. In our experiments, we used 480×480 images as inputs and outputted $120 \times 120 \times (K + 1)$ heatmaps per input. K is equal to 17 for COCO dataset. The lower resolutions harmed the mAP results while higher resolutions yielded longer training and inference complexity. We have listed the results of different keypoint models in Table 2.

Table 2. Comparison of different keypoint models. *(no concat: no concatenation, no int: no intermediate supervision, dil: dilated, concat: concatenation)*

Models	AP	AP$_{50}$	AP$_{75}$	AP$_M$	AP$_L$
R50	62.3	86.2	71.9	57.7	70.4
R101$_{no\ int.}$	61.3	83.7	69.6	56.6	67.4
R101$_{no\ concat}$	62.1	84.3	70.9	57.3	68.8
R101	63.9	87.1	73.2	58.1	72.2
R101$_{dil}$	**64.3**	**88.2**	**75**	**59.6**	**73.9**

The intermediate loss which is appended to the outputs of K blocks enhanced the precision significantly. Intermediate supervision acts as a refinement process among the hierarchies of features. As previously shown in [6, 38, 48], it is an essential strategy in most of the dense detection tasks. We have applied a final loss to the concatenated D features which is downsized from K features. This additional stage ensured us to combine multi-level features and compress them into a uniform space while extracting more semantic features. This strategy brought +2 mAP gain in our experiments.

Pose Residual Network Design PRN is a simple yet effective assignment strategy, and is designed for fast inference while giving reasonable accuracy. To design an accurate model we have tried different configurations. Different PRN models and corresponding results can be seen in Table 3. These results indicate

the scores obtained from the assignment of ground truth person bounding boxes and keypoints.

Table 3. Left: Performance of different PRN models on COCO validation set. *N: nodes, D: dropout and R: residual connection.* **Right:** Ablation experiments of PRN with COCO validation data.

PRN Models	AP	AP$_{50}$	AP$_{75}$	AP$_\text{M}$	AP$_\text{L}$
1 Layer 512 N, D	84.1	94.2	85.3	82	86.2
2 Layers 512 N, D	81.9	91.1	82.6	79.8	84.3
2 Layer 512 N, D+R	83.5	95.7	86.2	82.3	86.4
1 Layer 1024 N, D	84.6	95.7	87.6	82.1	88.7
1 Layer 1024 N, D+R	**89.4**	**97.1**	**91.2**	**87.9**	**91.8**

PRN Ablations	AP	AP$_{50}$	AP$_{75}$	AP$_\text{M}$	AP$_\text{L}$
Both GT	89.4	97.1	91.2	87.9	91.8
GT keypoints + Our bbox	75.3	82.1	78	70.1	84.5
Our keypoints + GT bbox	65.1	89.2	76.2	60.3	74.7
PRN	64.3	88.2	75	59.6	73.9
UCR	49.7	59.5	52.4	44.1	51.6
Max	45.3	55.1	48.8	40.6	46.9

We started with a primitive model which is a single hidden-layer MLP with 50 nodes, and added more nodes, regularization and different connection types to balance speed and accuracy. We found that 1024 nodes MLP, dropout with 0.5 probability and residual connection between input and output boosts the PRN performance up to 89.4 mAP on ground truth inputs.

In ablation analysis of PRN (Table 3), we compared *Max*, *UCR* and *PRN* implementations (see Section 3.4 for descriptions) along with the performance of PRN with ground truth detections. We found that lower order grouping methods could not handle overlapping detections; both of them performed poorly. As we hypothe-

Table 4. PRN assignment results with non-grouped keypoints obtained from two bottom-up methods.

Models	AP	AP$_{50}$	AP$_{75}$	AP$_\text{M}$	AP$_\text{L}$
Cao et al. [6]	58.4	81.5	62.6	**54.4**	65.1
PRN + [6]	**59.2**	**82.2**	**64.4**	54.1	**67.0**
Newell et al. [37]	56.9	80.8	61.3	49.9	**68.8**
PRN + [37]	**58.1**	**81.4**	**63.0**	**51.3**	68.1

sized, PRN could overcome ambiguities by learning meaningful pose structures (Fig. 2 (right)) and improved the results by ∼20 mAP over naive assignment techniques. We evaluated the impact of keypoint and person subnets to the final results by alternating inputs of PRN with ground truth detections. With ground truth keypoints and our person detections, we obtained 75.3 mAP, it shows that there is a large room for improvement in the keypoint localization part. With our keypoints and ground truth person detections, we obtained 65.1 mAP. This can be interpreted as our person detection subnet is performing quite well. Both ground truth detections got 89.4 mAP, which is a good indicator of PRN performance. In addition to these experiments, we tested PRN on the keypoints detected by previous SOTA bottom-up models [6, 37]. Consequently, PRN performed better grouping (Table 4) than *Part Affinity Fields* [6] and *Associative Embedding* [37] by improving both detection results by ∼1 mAP. To obtain results in Table 4, we have used COCO val split, our person bounding box results and the keypoint results from the official source code of the papers. Note that running PRN on keypoints that were not generated by MultiPoseNet is unfair to PRN because it is trained with our detection architecture. Moreover original methods use image features for assignment coupled with their detection scheme, nonetheless, PRN is able to outperform the other grouping methods.

4.3   Person Detection

We trained the person detection subnet only on COCO person instances by freezing the backbone with keypoint detection parameters. The person category results of our network with different backbones can be seen in Table 5. Our model with both ResNet-50 and ResNet-101 backends outperformed the original implementations. This is not a surprising result since our network is only dealing with a single class whereas the original implementations handle 80 object classes.

Table 5. Left: Person detection results on COCO dataset. **Right:**Person semantic segmentation results on PASCAL VOC 2012 test split.

Person Detectors	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L		Segmentation	IoU
Ours - R101	**52.5**	**81.5**	**55.3**	**35.2**	59	71		DeepLab v3 [9]	**92.1**
Ours - R50	51.3	81.4	53.6	34.9	58	68.1		DeepLab v2 [8]	87.4
RetinaNet [35]	50.2	77.7	53.5	31.6	**59**	**71.5**		SegNet [29]	74.9
FPN [34]	47.5	78	50.7	28.6	55	67.4		Ours	87.8

4.4   Person Semantic Segmentation

Person segmentation output is an additional layer appended to the keypoint outputs. We obtained the ground truth labels by combining person masks into single binary mask layer, and we jointly trained segmentation with keypoint task. Therefore, it adds a very small complexity to the model. Yet, producing segmentation masks didn't affect the keypoint results. Evaluation was performed on PASCAL VOC 2012 test set with PASCAL IoU metric. We obtained final segmentation results via multi-scale testing and thresholding. We did not apply any additional test-time augmentation or ensembling. Table 5 shows the test results of our system in comparison with previous successful semantic segmentation algorithms. Our model outperformed most of the successful baseline models such as SegNet [29] and Deeplab-v2 [8], and got comparable performance to the state-of-the-art Deeplab v3 [9] model. This demonstrates the capacity of our model to handle different tasks altogether with competitive performance. Some qualitative segmentation results are given in Fig. 5.

4.5   Runtime Analysis

Our system consists of a backbone, keypoint & person detection subnets, and the pose residual network. The parameter sizes of each block is given in the supplementary material. Most of the parameters are required to extract features in the backbone network, subnets and PRN are relatively lightweight networks. By using a shallow feature extractor like ResNet-50, we can achieve realtime performance. To measure the performance, we have built a model using ResNet-50 with 384×576 sized inputs which contain 1 to 20 people. We measured the time spent during the inference of 1000 images, and averaged the inference times to get a consistent result (Fig. 6). Keypoint and person detections take 35

Fig. 5. Some qualitative results for COCO test-dev dataset.

ms while PRN takes 2 ms per instance. Our model runs in betwen 15-27 FPS depending on the number of people in the image (15 FPS @ 1 person, 27 FPS @ 20 persons).

5   Conclusion

In this work, we introduced the Pose Residual Network which can accurately assign keypoints to person detections outputted by a multi task learning architecture (MultiPoseNet). Our pose estimation method achieved state-of-the-art performance among bottom-up methods and comparable results with top-down methods. Our method has the fastest inference time compared to previous methods. We showed the assignment performance of pose residual network ablation analysis. We

Fig. 6. Runtime analysis of MultiPoseNet with respect to number of people.

demonstrated the representational capacity of our multi-task learning model by jointly producing keypoints, person bounding boxes and person segmentation results.
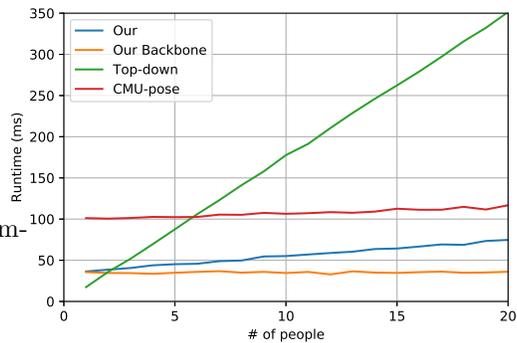
References

1. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Man, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Vigas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X.: Tensorflow: Large-scale machine learning on heterogeneous systems (2015), software available from tensorflow.org
2. Andriluka, M., Roth, S., Schiele, B.: Pictorial structures revisited: People detection and articulated pose estimation. In: IEEE Conference on Computer Vision and Pattern Recognition (2009)
3. Belagiannis, V., Zisserman, A.: Recurrent human pose estimation. In: International Conference on Automatic Face and Gesture Recognition (2017)
4. Bengio, Y., Louradour, J., Collobert, R., Weston, J.: Curriculum learning. In: International Conference on Machine Learning (2009)
5. Bulat, A., Tzimiropoulos, G.: Human pose estimation via convolutional part heatmap regression. In: European Conference on Computer Vision (2016)
6. Cao, Z., Simon, T., Wei, S.E., Sheikh, Y.: Realtime multi-person 2d pose estimation using part affinity fields. In: IEEE Conference on Computer Vision and Pattern Recognition (2017)
7. Carreira, J., Agrawal, P., Fragkiadaki, K., Malik, J.: Human pose estimation with iterative error feedback. In: IEEE Conference on Computer Vision and Pattern Recognition (2016)
8. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. In: IEEE Transaction on Pattern Analysis and Machine Intelligence (2017)
9. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: arXiv preprint arXiv:1802.02611 (2018)
10. Chen, Y., Wang, Z., Peng, Y., Zhang, Z., Yu, G., Sun, J.: Cascaded pyramid network for multi-person pose estimation. In: arXiv preprint arXiv:1711.07319 (2017)
11. Chollet, F., et al.: Keras (2015), https://github.com/keras-team/keras
12. Chou, C.J., Chien, J.T., Chen, H.T.: Self adversarial training for human pose estimation. In: arXiv preprint arXiv:1707.02439 (2017)
13. Chu, X., Yang, W., Ouyang, W., Ma, C., Yuille, A.L., Wang, X.: Multi-context attention for human pose estimation. In: IEEE Conference on Computer Vision and Pattern Recognition (2017)
14. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: IEEE Conference on Computer Vision and Pattern Recognition (2005)
15. Dantone, M., Gall, J., Leistner, C., Van Gool, L.: Human pose estimation using body parts dependent joint regressors. In: IEEE Conference on Computer Vision and Pattern Recognition (2013)
16. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition (2009)
17. Everingham, M., Eslami, S.M.A., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes challenge: A retrospective. In: International Journal of Computer Vision. vol. 111, pp. 98–136 (2015)

18. Fang, H., Xie, S., Tai, Y., Lu, C.: Rmpe: Regional multi-person pose estimation. In: International Conference on Computer Vision (2017)
19. Gaiser, H., de Vries, M., Williamson, A., Henon, Y., Morariu, M., Lacatusu, V., Liscio, E., Fang, W., Clark, M., Sande, M.V., Kocabas, M.: Keras-RetinaNet (2018), https://github.com/fizyr/keras-retinanet
20. Gkioxari, G., Arbelaez, P., Bourdev, L., Malik, J.: Articulated pose estimation using discriminative armlet classifiers. In: IEEE Conference on Computer Vision and Pattern Recognition (2013)
21. Gkioxari, G., Hariharan, B., Girshick, R., Malik, J.: Using k-poselets for detecting people and localizing their keypoints. In: IEEE Conference on Computer Vision and Pattern Recognition (2014)
22. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (2016)
23. He, K., Gkioxari, G., Dollr, P., Girshick, R.: Mask r-cnn. In: International Conference on Computer Vision (2017)
24. Huang, S., Gong, M., Tao, D.: A coarse-fine network for keypoint localization. In: International Conference on Computer Vision (2017)
25. Insafutdinov, E., Pishchulin, L., Andres, B., Andriluka, M., Schiele, B.: Deepercut: A deeper, stronger, and faster multi-person pose estimation model. In: European Conference on Computer Vision (2016)
26. Iqbal, U., Gall, J.: Multi-person pose estimation with local joint-to-person associations. In: European Conference on Computer Vision Workshops (2016)
27. Iqbal, U., Milan, A., Gall, J.: Posetrack: Joint multi-person pose estimation and tracking. In: IEEE Conference on Computer Vision and Pattern Recognition (2017)
28. Johnson, S., Everingham, M.: Clustered pose and nonlinear appearance models for human pose estimation. In: British Machine Vision Conference (2010)
29. Kendall, A., Badrinarayanan, V., , Cipolla, R.: Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. In: British Machine Vision Conference (2017)
30. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: International Conference on Learning Representations (2015)
31. Kong, T., Yao, A., Chen, Y., Sun, F.: Hypernet: Towards accurate region proposal generation and joint object detection. In: IEEE Conference on Computer Vision and Pattern Recognition (2016)
32. Ladicky, L., Torr, P.H., Zisserman, A.: Human pose estimation using a joint pixelwise and part-wise formulation. In: IEEE Conference on Computer Vision and Pattern Recognition (2013)
33. Lifshitz, I., Fetaya, E., Ullman, S.: Human pose estimation using deep consensus voting. In: European Conference on Computer Vision (2016)
34. Lin, T.Y., Dollr, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: IEEE Conference on Computer Vision and Pattern Recognition (2017)
35. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollr, P.: Focal loss for dense object detection. In: International Conference on Computer Vision (2017)
36. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollr, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European Conference on Computer Vision (2014)
37. Newell, A., Huang, Z., Deng, J.: Associative embedding: End-to-end learning for joint detection and grouping. In: Advances in Neural Information Processing (2017)
38. Newell, A., Yang, K., Deng, J.: Stacked hourglass networks for human pose estimation. In: European Conference on Computer Vision (2016)

39. Ning, G., Zhang, Z., He, Z.: Knowledge-guided deep fractal neural networks for human pose estimation. In: IEEE Transactions on Multimedia (2017)
40. Papandreou, G., Zhu, T., Kanazawa, N., Toshev, A., Tompson, J., Bregler, C., Murphy, K.: Towards accurate multi-person pose estimation in the wild. In: IEEE Conference on Computer Vision and Pattern Recognition (2017)
41. Pishchulin, L., Andriluka, M., Gehler, P., Schiele, B.: Poselet conditioned pictorial structures. In: IEEE Conference on Computer Vision and Pattern Recognition (2013)
42. Pishchulin, L., Insafutdinov, E., Tang, S., Andres, B., Andriluka, M., Gehler, P., Schiele, B.: Deepcut: Joint subset partition and labeling for multi person pose estimation. In: IEEE Conference on Computer Vision and Pattern Recognition (2016)
43. Ramakrishna, V., Munoz, D., Hebert, M., Bagnell, A.J., Sheikh, Y.: Pose machines: Articulated pose estimation via inference machines. In: European Conference on Computer Vision (2014)
44. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: IEEE Conference on Computer Vision and Pattern Recognition (2016)
45. Tompson, J., Jain, A., LeCun, Y., Bregler, C.: Joint training of a convolutional network and a graphical model for human pose estimation. In: Advances in Neural Information Processing (2014)
46. Toshev, A., Szegedy, C.: Deeppose: Human pose estimation via deep neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition (2014)
47. Varadarajan, S., Datta, P., Tickoo, O.: A greedy part assignment algorithm for realtime multi-person 2d pose estimation. In: arXiv preprint arXiv:1708.09182 (2017)
48. Wei, S.E., Ramakrishna, V., Kanade, T., Sheikh, Y.: Convolutional pose machines. In: IEEE Conference on Computer Vision and Pattern Recognition (2016)
49. Xia, F., Wang, P., Yuille, A., Angeles, L.: Joint multi-person pose estimation and semantic part segmentation in a single image. In: IEEE Conference on Computer Vision and Pattern Recognition (2017)
50. Xie, S., Girshick, R., Dollr, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition (2017)
51. Yang, Y., Ramanan, D.: Articulated pose estimation with flexible mixtures-of-parts. In: IEEE Transaction on Pattern Analysis and Machine Intelligence (2013)