

Learning Delicate Local Representations for Multi-Person Pose Estimation

Yuanhao Cai^{1,2*}, Zhicheng Wang^{1*}, Zhengxiong Luo^{1,3}, Binyi Yin^{1,4},
 Angang Du^{1,5}, Haoqian Wang², Xiangyu Zhang¹,
 Xinyu Zhou¹, Erjin Zhou¹, and Jian Sun¹

¹Megvii Inc. ²Tsinghua University ³Chinese Academy of Sciences
⁴Beihang University ⁵Ocean University of China

¹{*caiyuanhao, wangzhicheng, zxy, zej, zhangxiangyu, sunjian*}@megvii.com
²*wanghaoqian@tsinghua.edu.cn*

Abstract. In this paper, we propose a novel method called Residual Steps Network (RSN). RSN aggregates features with the same spatial size (Intra-level features) efficiently to obtain delicate local representations, which retain rich low-level spatial information and result in precise keypoint localization. Additionally, we observe the output features contribute differently to final performance. To tackle this problem, we propose an efficient attention mechanism - Pose Refine Machine (PRM) to make a trade-off between local and global representations in output features and further refine the keypoint locations. Our approach won the 1st place of COCO Keypoint Challenge 2019 and achieves state-of-the-art results on both COCO and MPII benchmarks, without using extra training data and pretrained model. Our single model achieves 78.6 on COCO test-dev, 93.0 on MPII test dataset. Ensembled models achieve 79.2 on COCO test-dev, 77.1 on COCO test-challenge dataset. The source code is publicly available for further research at <https://github.com/caiyuanhao1998/RSN/>

Keywords: Human Pose Estimation, COCO, MPII, Feature Aggregation, Attention Mechanism

1 Introduction

The goal of multi-person pose estimation is to locate keypoints of all persons in a single image. It is a fundamental task for human motion recognition, kinematics analysis, human-computer interaction, animation etc. For years, human pose estimation was based on handcraft features. Recently, It has made great progress with the development of deep convolutional neural network. The task of human pose estimation concerns both keypoint localization and classification. Spatial information benefits the localization task, while semantic information is good for the classification task. To extract these two kinds of information,

* The first two authors contribute equally to this work. This work is done when Yuanhao Cai, Zhengxiong Luo, Binyi Yin and Angang Du are interns at Megvii Research.

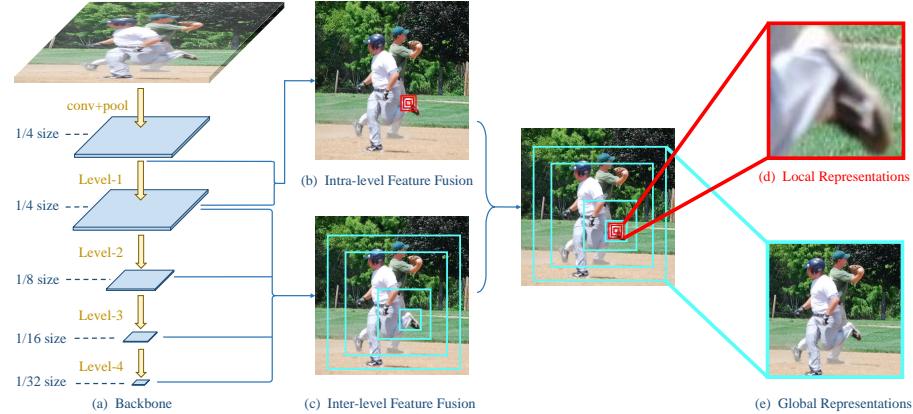


Fig. 1. Comparison of intra-level feature fusion and inter-level feature fusion. (a) Backbone. ”1/4 size” means 1/4 size of input image. (b) Intra-level feature fusion of level 1. (c) Inter-level feature fusion. (d) Local Representations. (e) Global representations.

current methods mainly focus on aggregating inter-level features. For instance, HRNet [24] maintains spatial information in high-resolution sub-network and gradually adds semantic information to it from low-resolution sub-networks. In this way, features of different levels are fully aggregated. In CPN [2], features of four different spatial levels are extracted by the backbone, and they are combined by a head network. Although these methods are different in the ways of feature fusion, the features to be aggregated are always from different levels. On the contrast, the feature fusion within the same level stays less explored in the task of human pose estimation.

The comparison of intra-level feature fusion (level 1) and inter-level feature fusion is illustrated in Figure 1. The feature maps are continuously downsampled to 1/4, 1/8, 1/16, 1/32 size of input image in Figure 1(a). We define consecutive feature maps with the same spatial size as one level. As Figure 1(c) depicts, there is a big gap between the receptive fields of features from different levels, which are indicated by light blue bounding boxes. As a result, representations learned by inter-level feature fusion are relatively coarse, which impede the localization of human pose from precise. As Figure 1(b) shows, the gap between the receptive fields of intra-level features which are indicated by red bounding boxes is relatively small. As shown in Figure 1(d), fusing intra-level features can extract much more delicate local representations retaining more precise spatial information, which is critical to keypoint localization.

To learn better local representations, we propose a novel network architecture - Residual Steps Network (RSN). The Residual Steps Block (RSB) of RSN fuses features inside each level using dense element-wise sum operations, which is shown in Figure 2(c). The inner structure of RSB is deeply connected and

motivated by DenseNet [10], which has a good performance for human pose estimation owing to retaining rich low-level features by deep connections. However, deep connections bring about explosion of the network capacity as it goes deeper. Thus, DenseNet performs poorly when the network becomes large. RSN is motivated by DenseNet but is quite different in that RSN uses element-wise sum rather than concatenation to circumvent network capacity explosion. RSN is modestly less dense connected in the block than DenseNet, which further promotes the efficiency. Additionally, we observe that the output features containing both global and local representations contribute differently to final performance. In light of this observation, we propose an attention module - Pose Refine Machine (PRM) to rebalance the output features of the network. The architecture of PRM is illustrated in Figure 3 and analyzed in Section 3.3. To better illustrate the advantages of our approach, we analyze the differences between RSN and current methods in Section 2.2.

In conclusion, our contributions can be summarized as three points:

1. We propose a novel network - RSN, which aims to learn delicate local representations by efficient intra-level feature fusion.
2. We propose an attention mechanism - PRM, which goes further to make a trade-off between local and global representations, and benefits the final performance.
3. Comprehensive experiments demonstrate that Our approach outperforms the state-of-the-art methods on both COCO and MPII datasets without using extra training data and pretrained model. Moreover, the proposed approach is much faster than HRNet with comparable performance on both GPU and CPU platforms.

2 Related Work

2.1 Multi-person Pose Estimation

Current methods of human pose estimation fall into two categories: top-down methods [20,11,8,5,24,17,23,22,19,2,29] and bottom-up methods [1,14,28,18]. Top-down methods first detect the positions of all persons, then estimate the pose of each person. Bottom-up methods first detect all the human keypoints in an image and then assemble these points into groups to form different individuals. Since this paper mainly concentrates on feature fusion strategies, we discuss these methods in terms of feature fusion.

2.2 Feature Fusion

Recently, many methods [17,29,2,31,24] of human pose estimation use inter-level feature fusion to extract more spatial and semantic information. Newell et al. [17] propose a U-shape convolutional neural network (CNN) named Hourglass. In a single stage of hourglass, high-level features are added to low-level features after upsampling. Later works such as Yang et al. [31] show great performance

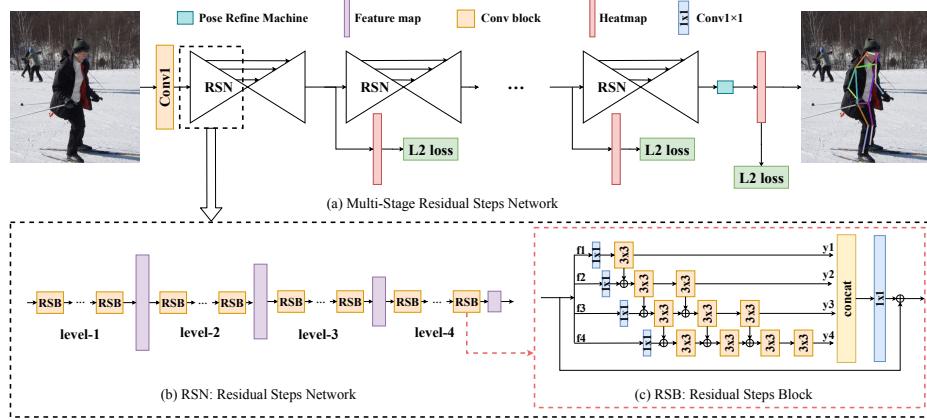


Fig. 2. Our pipeline. (a) is the multi-stage network architecture. It is cascaded by several Residual Steps Networks (RSNs). (b) is the backbone of RSN. (c) is the structure of Residual Steps Block (RSB), which is the basic block of RSN. RSB is designed for learning delicate local representations through dense element-wise sum connections. A Pose Refine Machine (PRM) is used in the last stage and it is analyzed in Section 3.4.

of using inter-level feature fusion. Chen et al. [2] combines inter-level features using a RefineNet. Sun et al. [24] set up four parallel sub-networks. The features of these four sub-networks aggregate with each other through high-to-low or low-to-high way.

Though many methods have validated the effectiveness of inter-level feature fusion, intra-level feature fusion is rarely explored in human pose estimation. However, it has extensive applications in other tasks such as semantic segmentation and image classification [25,7,10,4,34,30]. In a block of Inception [25], features pass through several convolutional layers with different kernels separately and then added up. DenseNet [10] fuses intra-level features using continuous concatenating operations. This implementation retains low-level features to improve the performance. However, when the network goes deeper, the capacity increases sharply and much redundant information appears in the network, resulting in poor efficiency. Different from DenseNet, RSN uses element-wise sum rather than concatenation to circumvent network capacity explosion. In addition, RSN is modestly less densely connected in the constituent unit, which further promotes the efficiency.

Res2Net [7] and OSNet [34] focus on multi-scale representations. Both of them lack dense connections between adjacent branches. The dense connections contribute sufficient gradients and make low-level features better supervised. Therefore, lack of dense connections between adjacent branches results in less precise spatial information, which is essential to keypoint localization. Suffering from this limitation, both Res2Net and OSNet are inferior to RSN in the task of human pose estimation. In Section 4.1, we validate the efficiency of DenseNet, Res2Net and RSN.

2.3 Attention Mechanism

Attention mechanism [27,9,23,13,32,6,33] is almost used in all areas of computer vision. Current methods of attention mechanism mainly fall into two categories: channel attention [27,9,23,32] and spatial attention [23,33,27,13,6]. Woo et al. [27] propose a channel attention module with global average pooling and max pooling. Kligvasser et al. [13] propose a spatial activation function with depth-wise separable convolution. Other works such as Hu et al. [9] show the advantages of using attention mechanism. However, most prior attention modules are lack of representing capacity and focus on optimizing the backbone. We design PRM to make a trade-off between local and global representations in output features by using powerful while computation-economical operations.

3 Proposed Method

The overall pipeline of our method is illustrated in Figure 2. The multi-stage network architecture is cascaded by several single-stage modules - Residual Steps Network (RSN), shown in Figure 2(a). As Figure 2(b) shows, RSN differs from ResNet in the architecture of constituent unit. RSN consists of Residual Steps Blocks (RSBs) while ResNet is comprised of "bottleneck" blocks. Figure 2(c) illustrates the structure of RSB. A Pose Refine Machine (PRM) is used in the last stage and it is analyzed in Section 3.3.

3.1 Delicate Local Representations Learning

Residual Steps Network is designed for learning delicate local representations by repeatedly enhancing efficient intra-level feature fusion inside RSB, which is the constituent unit of RSN. As shown in Figure 2(c), RSB firstly divides the features into four splits f_i ($i = 1, 2, 3, 4$), then implements a $\text{conv}1 \times 1$ (convolutional layer with kernel size 1×1) separately. Each feature output from $\text{conv}1 \times 1$ undergoes incremental numbers of $\text{conv}3 \times 3$. The output features y_i ($i = 1, 2, 3, 4$) are then concatenated to go through a $\text{conv}1 \times 1$. An identity connection is employed as the ResNet bottleneck. Because the incremental numbers of $\text{conv}3 \times 3$ look like steps, the network is therefore named Residual Steps Network.

The receptive fields of RSB range across several values, and the max one is 15. Compared with a single receptive field in ResNet bottleneck as shown in Table 1, RSB indicates more delicate information viewed in the network. In addition, it is deeply connected inside RSB. On the i^{th} branch, the front $i - 1$ $\text{conv}3 \times 3$ receive the features output from the $(i - 1)^{\text{th}}$ branch. The i^{th} $\text{conv}3 \times 3$ is then designed to refine the fusion of the features output from the $(i - 1)^{\text{th}}$ $\text{conv}3 \times 3$. Benefit from the dense connection structure, small-gap receptive fields of features are fully fused resulting in delicate local representations, which retain precise spatial and semantic information. Additionally, during the training process, the deeply connected structure contributes sufficient gradients, so the low-level features are better supervised, which benefits the keypoint localization task. We investigate how the branch number of RSB influences the prediction results in Section 4.1. Four-branch architecture has the best performance.

Table 1. The receptive field comparison between RSB and other methods.

Architecture	y1	y2	y3	y4
ResNet	3	3	3	3
OSNet	3	5	7	9
Res2Net	1	3	3,5	3,5,7
RSN	3	5,7	7,9,11	9,11,13,15

3.2 Receptive Field Analysis

In this part, we analyze the receptive fields in RSB and other methods. Firstly, the formula for calculating the receptive field of the k^{th} convolutional layer is written as Equation 1

$$l_k = l_{k-1} + [(f_k - 1) * \prod_{i=1}^{k-1} s_i] \quad (1)$$

l_k denotes the size of the receptive field corresponding to the k^{th} layer, f_k denotes the kernel size of the k^{th} layer and s_i denotes the stride of the i^{th} layer. In this part, we only focus on the change of relative receptive fields in a block. Every f_k is 3 and s_i is 1. Thus, Equation 1 can be simplified to Equation 2

$$l_k = l_{k-1} + 2 \quad (2)$$

Using this formula, we calculate the relative receptive fields of RSB and other methods, as shown in Table 1. It indicates that RSN has a wider range of scales than ResNet, Res2Net and OSNet. The scale of each human joint varies a lot. For instance, the scale of eye is small while that of hip is large. For this reason, architecture with wider range of receptive fields is more fit for extracting features relating to different joints. Also, wider range of receptive fields helps to learn more discriminant semantic representations, which benefits the keypoint classification task. More importantly, RSN builds dense connections between the features with small-gap receptive fields inside RSB. The deeply connected architecture contributes to learning delicate local representations, which are essential to precise human pose estimation.

3.3 Pose Refine Machine

In the last module of multi-stage network (Figure 2(a)), an attention mechanism - Pose Refine Machine (PRM) is used to reweight the output features, as shown in Figure 3. The first component of PRM is a conv3×3, then the features are input into three paths. The top path is an identity connection. The middle one, motivated by SENet [9], passes through a global pooling, two conv1×1 and a sigmoid activation to get a weight vector α . The bottom path passes through a conv1×1, a depth-wise separable conv9×9 and a sigmoid activation to get an attention map β . Element-wise sum and multiplication are conducted among

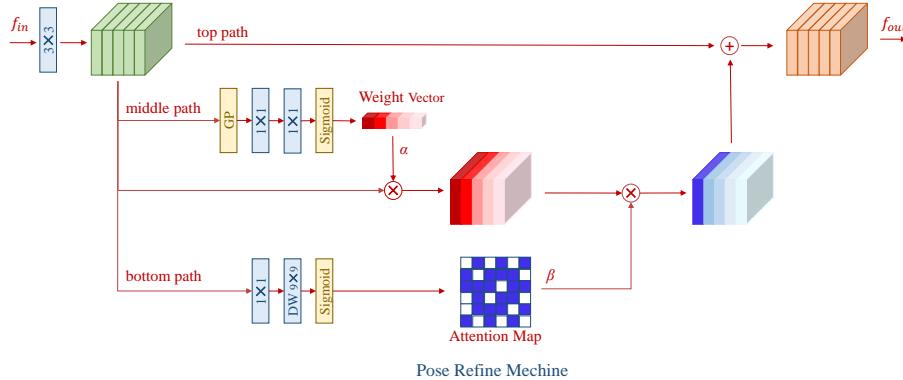


Fig. 3. Architecture of Pose Refine Machine (PRM). GP denotes global pooling. DW denotes depth-wise separable convolution. α denotes the weight vector. β denotes the attention map. The top path is an identity connection, the middle path is designed to reweight features in channel wise and the bottom path is proposed for spatial attention.

the three paths to get the output features. Define the input features of PRM as f_{in} , the output features as f_{out} , the first conv 3×3 as $K(\cdot)$ and element-wise multiplication as \odot . Then PRM can be formulated as Equation 3.

$$f_{out} = K(f_{in}) \odot (1 + \beta \odot \alpha) \quad (3)$$

As for the output of RSN, features after intra-level and inter-level aggregation are mixed together containing both low-level precise spatial information and high-level discriminant semantic information. Spatial information is good for keypoint localization while semantic information benefits keypoints classification. These features contribute differently to the final prediction. Therefore, to tackle this imbalance problem, PRM is designed to make a trade-off between local and global representations in output features of RSN. Compared to prior work of attention mechanism, we use powerful while computation-economical operations, e.g. conv 3×3 , conv 1×1 and DW conv 9×9 . The top identity mapping in PRM is good for retaining local features which benefits precise keypoint localization. The middle path is designed to reweight the features in channel wise and the bottom path is proposed for spatial attention.

4 Experiments

4.1 COCO Keypoints Detection

Datasets, Evaluation Metric, Human Detection. COCO dataset [16] includes over 200K images and 250K person instances labeled with 17 joints. We use only COCO train2017 dataset for training (including about 57K images and 150K person instances). We evaluate our method on COCO minival

dataset (5K images) and the testing datasets including test-dev (20K images) and test-challenge (20K images). We use standard OKS-based AP score as the evaluation metric. We use MegDet and MegDet-v2 as human detector on COCO val and test sets respectively.

Training Details. The network is trained on 8 V100 GPUs with mini-batch size 48 per GPU. There are 140k iterations per epoch and 200 epochs in total. Adam optimizer is adopted and the linear learning rate gradually decreases from 5e-4 to 0. The weight decay is 1e-5. Each image goes through a series of data augmentation operations including cropping, flipping, rotation and scaling. The range of rotation is $-45^\circ \sim +45^\circ$. The range of scaling is 0.7~1.35. The size of input image is 256×192 or 384×288 .

Testing Details. We apply a post-Gaussian filter to the estimated heatmaps. Following the strategy of hourglass [17], we average the predicted heatmaps of original image with the results of corresponding flipped image. Then we implement a quarter offset from the highest response to the second highest one to get the locations of keypoints. The same with CPN [2], the pose score is the multiplication of the average score of keypoints and the bounding box score.

Table 2. Results of ResNet, Res2Net, Baseline1,2 and RSN on COCO val set

backbone	input size	AP	Δ	GFLOPs
ResNet-18	256×192	70.7	0	2.3
Res2Net-18	256×192	71.3	+0.6	2.2
Baseline1-18	256×192	72.9	+2.1	2.5
Baseline2-18	256×192	72.1	+1.4	2.5
RSN-18	256×192	73.6	+2.9	2.5
ResNet-50	256×192	72.2	0	4.6
Res2Net-50	256×192	72.8	+0.6	4.5
Baseline1-50	256×192	73.7	+1.5	6.4
Baseline2-50	256×192	72.7	+0.5	6.4
RSN-50	256×192	74.7	+2.5	6.4
ResNet-101	256×192	73.2	0	7.5
Res2Net-101	256×192	73.9	+0.7	7.5
RSN-101	256×192	75.8	+2.5	11.5
$4 \times$ ResNet-50	256×192	76.8	0	20.6
$4 \times$ Res2Net-50	256×192	77.0	+0.2	20.1
$4 \times$RSN-50	256×192	78.6	+1.8	27.5
$4 \times$ ResNet-50	384×288	77.5	0	46.4
$4 \times$ Res2Net-50	384×288	77.6	+0.1	45.2
$4 \times$RSN-50	384×288	79.2	+1.7	61.9

Ablation Study of RSN Improvement. In Section 2.2, we analyze the differences between RSN and current methods. In this part, we validate the effectiveness of intra-level feature fusion method in RSN. Since it is known dividing the

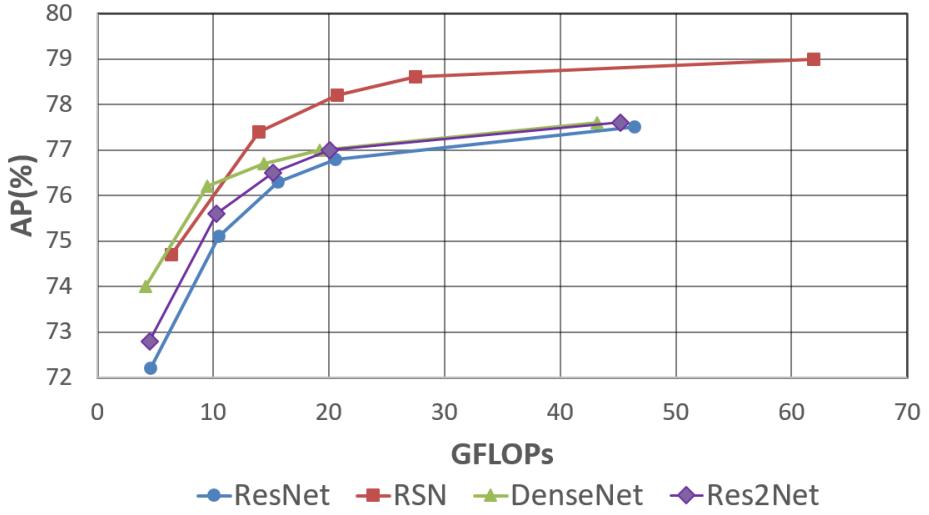


Fig. 4. Illustrating how the performances of ResNet, Res2Net, DenseNet and RSN are affected by GFLOPs. The results are reported on COCO minival dataset.

network into branches, e.g., Inception and ResNetXt [30], can improve the recognition performance, we add two baselines into comparison. Baseline 1: remove the intra-level fusion (i.e., the vertical arrows) from Figure 2(c). This baseline can reveal whether the proposed intra-level fusion is important. Baseline 2: replace f1-f4 in Baseline 1 with 4 f3's respectively, which is more like the conventional branching strategy. We keep the same GFLOPs of Baseline1, Baseline2 with RSN by adapting channels. Ablation experiments are implemented on ResNet, Res2Net, Baseline1, Baseline2, and RSN based networks. PRM is left out for more strong comparison. The results on COCO val are reported in Table 2.

As Table 2 shows, RSN boosts the performance by relatively larger extent with acceptable computation cost addition, while Res2Net can only obtain limited gain. For instance, RSN-18 is 2.9 points AP higher than ResNet-18 adding only 0.2 GFLOPs and 2.3 points AP higher than Res2Net-18 adding only 0.3 GFLOPs. However, Res2Net-18 obtains only 0.6 AP gain than ResNet-18. RSN always works much better than ResNet and Res2Net with comparable GFLOPs. In addition, it is worth noting that when model complexity is relatively low, RSN still has a remarkable performance, which indicates that RSN is more compact and efficient. For instance, compared with ResNet-101 and Res2Net-101, RSN-18 has a similar AP, however, with only a third of computation cost. On the other hand, RSN achieves higher AP than Baseline1 and Baseline2 with the same GFLOPs, e.g., RSN-50 is 1 AP higher than Baseline1-50 and 2 AP higher than Baseline2-50. This observation strongly demonstrate the superiority of the intra-level feature fusion mode of RSN.

Ablation Study of RSN Efficiency. The dense connection principle of RSN comes from DenseNet. However, it is not efficient for DenseNet when too many concatenating operations are implemented. To circumvent the network capacity explosion, RSN uses element-wise sum to connect adjacent branches. To validate the efficiency of our approach, we respectively adopt ResNet, Res2Net, DenseNet and RSN as the backbone in the same multi-stage architecture as shown in Figure 2(a) to compare the performance. PRM is left out for fair comparison. The results are shown in Figure 4. For relatively small models, RSN and DenseNet based networks can both achieve good results, while Res2Net only gets a minor improvement than ResNet. However, as the model capacity increases, the improvements of DenseNet and Res2Net based network decrease dramatically. Both of them can only get an inferior result close to ResNet when the model size becomes large, while RSN can keep its superiority to the end.

DenseNet has a high AP score at a low complexity owing to the deep connections and frequent feature aggregations inside the same level by continuous concatenating operations. This makes the low-level features sufficiently supervised resulting in satisfactory delicate spatial texture information, which benefits the keypoint localization. However, as the computation cost raises, the concatenating operations of DenseNet become redundant. It combines quite a large range of less utilized information. As for Res2Net, narrower range of receptive fields and lack of efficient intra-level feature fusion to extract delicate local representations make it much inferior than RSN.

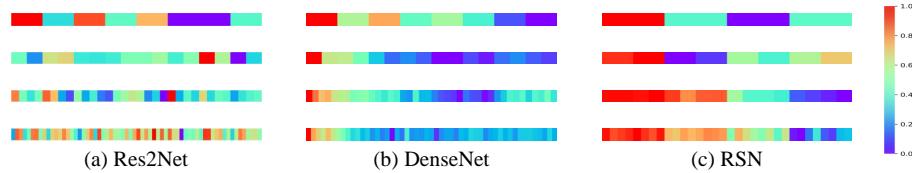


Fig. 5. The average absolute filter weights of the last $conv1 \times 1$ layers of each level in trained Res2Net-50 (a), DenseNet-169(b) and RSN-50(c). Larger weights means higher utilization. The weights of Res2Net are smaller than those of RSN. Most weights in DenseNet have values close to zero. While RSN can utilize most channels better.

In order to embody the differences of Res2Net, DenseNet and RSN more essentially, we show the average absolute filter weights of the last $conv1 \times 1$ layers of each level in trained Res2Net-50, DenseNet-169 and RSN-50 in Figure 5. The highly used weights become less from level 1 to level 4 in DenseNet. The overall useful weights of DenseNet are less than those of RSN, which can be deduced from Figure 5 (b) and (c) that the red area in each level of DenseNet is much smaller than that of RSN. According to the analysis in Section 3.2, RSN can enhance the efficient fusion of intra-level features with dense element-wise sum connections. There are not accumulative concatenating operations like DenseNet. Thus, RSN is less occupied by the redundant features with low utilization. On the

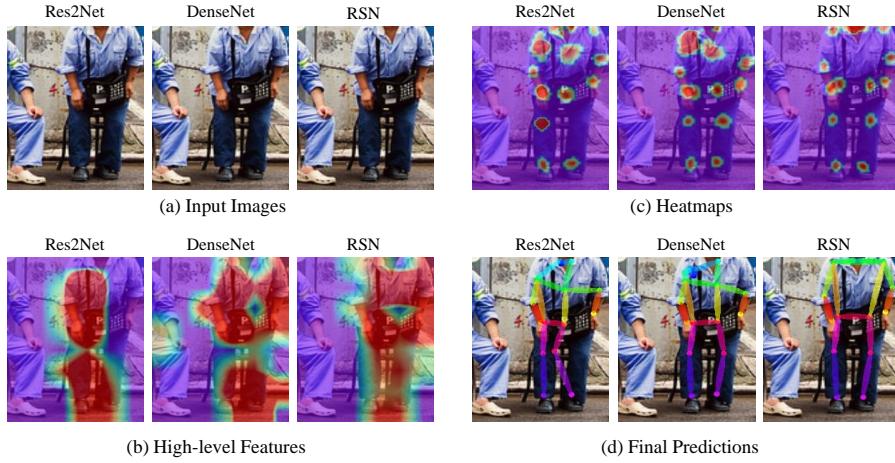


Fig. 6. Visual analysis of Res2Net-50, DenseNet-169 and RSN-50. (a) Input images, (b) High-level feature maps (level 4), (c) Low-level heatmaps (level 1), (d) Final predictions.

other hand, compared with Res2Net, the more densely connected architecture and wider range of receptive fields make the intra-level feature fusion of RSN more effective, that is why the red area of RSN is much larger than that of Res2Net and the weights of RSN are more fully used, just as shown in Figure 5 (a) and (c). As a result, the RSN model can keep its high efficiency and considerable improvement from the beginning to the end, just as shown in Figure 4.

Additionally, to highlight the advantages of RSN more intuitively, we conduct visual analysis of Res2Net-50, DenseNet-169 and RSN-50, as shown in Figure 6. In Figure 6(b), the high-level response to human body of Res2Net and DenseNet either covers incomplete body area or too large area of background. Only RSN has a relatively complete and appropriate response area to the human body. As a result, in final prediction, Res2Net is easily misled by the background information, DenseNet ignores some keypoints such as shoulders, while RSN can locate the keypoints better and reduce the interference of background information. As Figure 6(c) shows, the heatmaps of RSN are much clearer and the locations of the responses are much more accurate.

Ablation Study of RSN Architecture. When designing RSN, we firstly deploy the dense connection principle of DenseNet. Then, for a break-down ablation, we set different branch number as variants to discuss the designing of RSN and explore a best trade-off between branch representing capacity and the degree of intra-level fusion. experiments are done on RSN-18 and RSN-50. We increase branch numbers from 2 to 6 while keeping the model capacity unchanged by adapting channels. As Table 3 shows, the performance firstly becomes better and attains its peak when there are 4 branches. However, when the branch number continues growing, the results get worse. Thus, 4 is the best choice.

Table 3. Illustrating how the performance of RSN affected by the branch number.

backbone	input size	2-branch	3-branch	4-branch	5-branch	6-branch
RSN-18	256×192	73.1	73.0	73.6	73.2	72.9
RSN-50	256×192	73.9	74.2	74.7	74.3	74.0

Ablation Study of Pose Refine Machine. In Section 3.3, we have analyzed the architecture of PRM and the differences between PRM and prior attention mechanism. To validate the improvement of PRM, we perform ablation experiments on both single-stage and multi-stage network architecture. Additionally, We validate the impact of SENet and CBAM by replacing PRM. The results are shown in Table 4. SE-block and CBAM decrease the performance of human pose estimation, which implies vanilla attention mechanisms are not suitable for rebalancing output features. In contrast, when the model capacity is small, PRM has a considerable improvement. As for relatively high AP baseline, PRM still obtains 0.4 AP gain. These observations demonstrate the robustness of PRM.

Results on COCO test-dev and test-challenge. We validate our approach on COCO test-dev and test-challenge sets. The results are shown in Table 5 and Table 6. For fair comparison, we pay attention to the performances of single models with comparable GFLOPs, without using extra training data. In Table 5, our method outperforms HRNet by 2.5 AP (78.0 v.s. 75.5), and outperforms SimpleBaseline by 4.3 AP on COCO test-dev dataset. Additionally, as Table 6 shows, our approach outperforms MSPN (winner of COCO kps Challenge 2018) by 0.7 AP on test-challenge set. Note that we don’t even use pretrained model.

Table 4. Ablation experiments of Pose Refine Machine on COCO minival dataset.

Backbone	Attention	input size	AP	Δ	GFLOPs
ResNet-18	None	256×192	70.7	0	2.3
ResNet-18	SE-block	256×192	70.5	-0.2	2.3
	CBAM	256×192	69.9	-0.8	2.3
	PRM	256×192	72.2	+1.5	4.1
	None	256×192	72.2	0	4.6
ResNet-50	SE-block	256×192	72.1	-0.1	4.6
	CBAM	256×192	71.1	-1.1	4.6
	PRM	256×192	73.4	+1.2	6.4
	None	256×192	76.8	0	20.6
4×ResNet-50	SE-block	256×192	76.6	-0.2	20.6
	CBAM	256×192	76.1	-0.7	20.6
	PRM	256×192	77.2	+0.4	22.4
	None	256×192	78.6	0	27.5
4×RSN-50	SE-block	256×192	78.6	0	27.5
	CBAM	256×192	78.0	-0.6	27.5
	PRM	256×192	79.0	+0.4	29.3

Table 5. Results on COCO test-dev dataset. “**” means using ensembled models. Pretrain = pretrain the backbone on the ImageNet classification task.

Method	Extra data	Pretrain	Backbone	Input Size	Params	GFLOPs	AP	AP.5	AP.75	AP(M)	AP(L)	AR
CMUpose [1]	x	-	-	-	-	-	61.8	84.9	67.5	57.1	68.2	66.5
G-MRI [21]	x	-	ResNet-101	353×257	42.6M	57.0	64.9	85.5	71.3	62.3	70.0	69.7
G-RMI [21]	✓	-	ResNet-101	353×257	42.6M	57.0	68.5	87.1	75.5	65.8	73.3	73.3
CPN [2]	x	✓	ResNet-Inception	384×288	58.8M	29.2	72.1	91.4	80.0	68.7	77.2	78.5
CPN* [2]	x	✓	ResNet-Inception	384×288	-	-	73.0	91.7	80.9	69.5	78.1	79.0
SimpleBase [29]	x	✓	ResNet-152	384×288	68.6M	35.6	73.7	91.9	81.1	70.3	80.0	79.0
HRNet-W32 [24]	x	✓	HRNet-W32	384×288	28.5M	16.0	74.9	92.5	82.8	71.3	80.9	80.1
HRNet-W48 [24]	x	✓	HRNet-W48	384×288	63.6M	32.9	75.5	92.5	83.3	71.9	81.5	80.5
SimpleBase* [29]	✓	✓	ResNet-152	384×288	-	-	76.5	92.4	84.0	73.0	82.7	81.5
HRNet-W48 [24]	✓	✓	HRNet-W48	384×288	63.6M	32.9	77.0	92.7	84.5	73.4	83.1	82.0
MSPN [15]	✓	x	4×ResNet-50	384×288	71.9M	58.7	77.1	93.8	84.6	73.4	82.3	82.3
Ours(RSN)	x	x	RSN-18	256×192	12.5M	2.5	71.6	92.6	80.3	68.8	75.8	77.7
Ours(RSN)	x	x	RSN-50	256×192	25.7M	6.4	72.5	93.0	81.3	69.9	76.5	78.8
Ours(RSN)	x	x	2×RSN-50	256×192	54.0M	13.9	75.5	93.6	84.0	73.0	79.6	81.3
Ours(RSN)	x	x	4×RSN-50	256×192	111.8M	29.3	78.0	94.2	86.5	75.3	82.2	83.4
Ours(RSN)	x	x	4×RSN-50	384×288	111.8M	65.9	78.6	94.3	86.6	75.5	83.3	83.8
Ours(RSN*)	x	x	4×RSN-50	-	-	-	79.2	94.4	87.1	76.1	83.8	84.1

Inference Speed. Current methods of human pose estimation mainly focus on promoting the performance while deploying resource-intensive networks with large depth and width. This leads to inefficient inference. Interestingly, we observe RSN can make a better trade-off between accuracy and inference speed than prior work. For fair comparison, we train RSN and HRNet under the same settings in Section 4, with 256×192 input size. Both use MegDet as human detector when testing. We use pps to measure inference speed, i.e., Persons inferred Per Second. On the same GPU (RTX 2080ti), results of COCO val are reported, HRNet-w16 with 1.9 G and 7.2 M achieves 71.9 AP and 31.8 pps, RSN-18 with 2.5G and 12.5 M achieves 73.6 AP and 64.9 pps, HRNet-w32 with 7.1 G and 28.5M achieves 74.6 AP and 26.5 pps, RSN-50 with 6.4 G and 25.7 M achieves 74.7 AP and 42.6 pps. HRNet-w48 with 14.6G and 63.6M achieves 75.5 AP and 24.7 pps, $2 \times$ RSN-50 with 13.9 G and 54.0 M achieves 77.4 AP and 20.2 pps. In addition, the inference speed on CPU (Intel(R) Xeon(R) Gold6013@2.1GHZ) also shows, RSNs with higher performances are faster than HRNet by all sizes. These results suggest that RSN is more accurate, compact and efficient.

Effect of Human Detection. We use MegDet as human detector in ablation study, which achieves 49.4 AP on COCO val. For test sets, we use MegDet-v2, which has 59.8 AP on COCO val. As human detection has an influence on the final performance of top-down approach, We perform ablations to investigate the impact of human detector on COCO test-dev. $4 \times$ RSN-50 at input size of 256×192 achieves 77.3 AP using MegDet, and 78.0 using MegDet-v2. $4 \times$ RSN-50 at input size of 384×288 achieves 77.9 using MegDet, and 78.6 using MegDet-v2.

4.2 MPII Human Pose Estimation

We validate RSN on MPII test set, a single-person pose estimation benchmark. As shown in Table 7, RSN boosts the SOTA performance by 0.7 in PCKh@0.5, which demonstrates the superiority and generalization ability of our method.

Table 6. Results on COCO test-challenge dataset. “**” means using ensembled models.

Method	Extra data	Pretrain	Backbone	Input Size	Params	GFLOPs	AP	AP.5	AP.75	AP(M)	AP(L)	AR
G-RMI [21]	✓	-	ResNet-101	353×257	42.6M	57.0	69.1	85.9	75.2	66.0	74.5	75.1
CPN* [2]	✗	✓	ResNet-Inception	384×288	-	-	72.1	90.5	78.9	67.9	78.1	78.7
Sea Monsters*	✓	-	-	-	-	-	74.1	90.6	80.4	68.5	82.1	79.5
SimpleBase* [29]	✓	✓	ResNet-152	384×288	-	-	74.5	90.9	80.8	69.5	82.9	80.5
MSPN* [15]	✓	✗	4×ResNet-50	384×288	-	-	76.4	92.9	82.6	71.4	83.2	82.2
Ours(RSN*)	✗	✗	4×RSN-50	-	-	-	77.1	93.3	83.6	72.2	83.6	82.6

Table 7. PCKh@0.5 results on MPII test dataset.

Method	Hea	Sho	Elb	Wri	Hip	Kne	Ank	Mean
Chen et al.[3]	98.1	96.5	92.5	88.5	90.2	89.6	86.0	91.9
Yang et al.[31]	98.5	96.7	92.5	88.7	91.1	88.6	86.0	92.0
Ke et al. [12]	98.5	96.8	92.7	88.4	90.6	89.3	86.3	92.1
Tang et al. [26]	98.4	96.9	92.6	88.7	91.8	89.4	86.2	92.3
Sun et al. [24]	98.6	96.9	92.8	89.0	91.5	89.0	85.7	92.3
ours(4×RSN-50)	98.5	97.3	93.9	89.9	92.0	90.6	86.8	93.0

5 Conclusion

In this paper, we propose a novel method, Residual Steps Network, which aims to learn delicate local representations by efficient intra-level feature fusion. To make a better trade-off between local and global representations in output features, we design Pose Refine Machine. Our method yields the best results on two benchmarks, COCO and MPII. Some prediction results are visualized in Fig 7.

**Fig. 7.** Prediction results on COCO (top line) and MPII (bottom line) val sets.

6 Acknowledgement

This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFA0700800.

References

1. Cao, Z., Simon, T., Wei, S.E., Sheikh, Y.: Realtime multi-person 2d pose estimation using part affinity fields. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (July 2017)
2. Chen, Y., Wang, Z., Peng, Y., Zhang, Z., Yu, G., Sun, J.: Cascaded pyramid network for multi-person pose estimation. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018)
3. Chen, Y., Shen, C., Wei, X.S., Liu, L., Yang, J.: Adversarial posenet: A structure-aware convolutional network for human pose estimation. In: The IEEE International Conference on Computer Vision (ICCV) (Oct 2017)
4. Chen, Y., Li, J., Xiao, H., Jin, X., Yan, S., Feng, J.: Dual path networks. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems 30, pp. 4467–4475. Curran Associates, Inc. (2017), <http://papers.nips.cc/paper/7033-dual-path-networks.pdf>
5. Fang, H.S., Xie, S., Tai, Y.W., Lu, C.: Rmpe: Regional multi-person pose estimation. In: The IEEE International Conference on Computer Vision (ICCV) (Oct 2017)
6. Fu, J., Liu, J., Tian, H., Li, Y., Bao, Y., Fang, Z., Lu, H.: Dual attention network for scene segmentation. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019)
7. Gao, S.H., Cheng, M.M., Zhao, K., Zhang, X.Y., Yang, M.H., Torr, P.: Res2net: A new multi-scale backbone architecture. In: IEEE TPAMI (2020)
8. He, K., Gkioxari, G., Dollar, P., Girshick, R.: Mask r-cnn. In: The IEEE International Conference on Computer Vision (ICCV) (Oct 2017)
9. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018)
10. Huang, G., Liu, Z., van der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (July 2017)
11. Huang, S., Gong, M., Tao, D.: A coarse-fine network for keypoint localization. In: The IEEE International Conference on Computer Vision (ICCV) (Oct 2017)
12. Ke, L., Chang, M.C., Qi, H., Lyu, S.: Multi-scale structure-aware network for human pose estimation. In: The European Conference on Computer Vision (ECCV) (September 2018)
13. Kligvasser, I., Rott Shaham, T., Michaeli, T.: xunit: Learning a spatial activation function for efficient image restoration. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018)
14. Kocabas, M., Karagoz, S., Akbas, E.: Multiposenet: Fast multi-person pose estimation using pose residual network. In: The European Conference on Computer Vision (ECCV) (September 2018)
15. Li, W., Wang, Z., Yin, B., Peng, Q., Du, Y., Xiao, T., Yu, G., Lu, H., Wei, Y., Sun, J.: Rethinking on multi-stage networks for human pose estimation. arXiv preprint arXiv:1901.00148 (2019)
16. Lin, T., Maire, M., Belongie, S.J., Hays, J., Perona, P., Ramanan, D., Dollr, P., Zitnick., C.L.: Microsoft coco: common objects in context. In: ECCV. pp. 740–755 (June 2014)
17. Newell, A., Yang, K., Deng., J.: Stacked hourglass networks for human pose estimation. In: The European Conference on Computer Vision (ECCV) (2016)

18. Newell, A., Huang, Z., Deng, J.: Associative embedding: End-to-end learning for joint detection and grouping. In: Advances in Neural Information Processing Systems 30. pp. 2277–2287 (2017)
19. nsafutdinov, E., Pishchulin, L., Andres, B., Andriluka, M., Schiele, B.: Deepcut: A deeper, stronger, and faster multi-person pose estimation model. In: ECCV (2016)
20. Papandreou, G., Zhu, T., Kanazawa, N., Toshev, A., Tompson, J., Bregler, C., Murphy, K.: Towards accurate multi-person pose estimation in the wild. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (July 2017)
21. Papandreou, G., Zhu, T., Kanazawa, N., Toshev, A., Tompson, J., Bregler, C., Murphy, K.: Towards accurate multi-person pose estimation in the wild. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (July 2017)
22. Pishchulin, L., Insafutdinov, E., Tang, S., Andres, B., Andriluka, M., Gehler, P.V., Schiele, B.: Deepcut: Joint subset partition and labeling for multi person pose estimation. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2016)
23. Su, K., Yu, D., Xu, Z., Geng, X., Wang, C.: Multi-person pose estimation with enhanced channel-wise and spatial information. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019)
24. Sun, K., Xiao, B., Liu, D., Wang, J.: Deep high-resolution representation learning for human pose estimation. In: CVPR (2019)
25. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2015)
26. Tang, W., Yu, P., Wu, Y.: Deeply learned compositional models for human pose estimation. In: The European Conference on Computer Vision (ECCV) (September 2018)
27. Woo, S., Park, J., Lee, J.Y., So Kweon, I.: Cbam: Convolutional block attention module. In: The European Conference on Computer Vision (ECCV) (September 2018)
28. Xia, F., Wang, P., Chen, X., Yuille, A.L.: Joint multi-person pose estimation and semantic part segmentation. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (July 2017)
29. Xiao, B., Wu, H., Wei, Y.: Simple baselines for human pose estimation and tracking. In: The European Conference on Computer Vision (ECCV) (September 2018)
30. Xie, S., Girshick, R., Dollar, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (July 2017)
31. Yang, W., Li, S., Ouyang, W., Li, H., Wang, X.: Learning feature pyramids for human pose estimation. In: The IEEE International Conference on Computer Vision (ICCV) (Oct 2017)
32. Yu, C., Wang, J., Peng, C., Gao, C., Yu, G., Sang, N.: Learning a discriminative feature network for semantic segmentation. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018)
33. Zhao, H., Zhang, Y., Liu, S., Shi, J., Change Loy, C., Lin, D., Jia, J.: Psanet: Pointwise spatial attention network for scene parsing. In: The European Conference on Computer Vision (ECCV) (September 2018)
34. Zhou, K., Yang, Y., Cavallaro, A., Xiang, T.: Omni-scale feature learning for person re-identification. In: ICCV (2019)