Multi-Person Pose Estimation with Enhanced Channel-wise and Spatial Information

Kai Su[†,1,2], Dongdong Yu[†,2], Zhenqi Xu[2], Xin Geng[*,1], Changhu Wang[*,2]
[1]School of Computer Science and Engineering, Southeast University, Nanjing, China
{sukai,xgeng}@seu.edu.cn
[2]ByteDance AI Lab, Beijing, China
{sukai,yudongdong,xuzhenqi,wangchanghu}@bytedance.com

Abstract

Multi-person pose estimation is an important but challenging problem in computer vision. Although current approaches have achieved significant progress by fusing the multi-scale feature maps, they pay little attention to enhancing the channel-wise and spatial information of the feature maps. In this paper, we propose two novel modules to perform the enhancement of the information for the multi-person pose estimation. First, a Channel Shuffle Module (CSM) is proposed to adopt the channel shuffle operation on the feature maps with different levels, promoting cross-channel information communication among the pyramid feature maps. Second, a Spatial, Channel-wise Attention Residual Bottleneck (SCARB) is designed to boost the original residual unit with attention mechanism, adaptively highlighting the information of the feature maps both in the spatial and channel-wise context. The effectiveness of our proposed modules is evaluated on the COCO keypoint benchmark, and experimental results show that our approach achieves the state-of-the-art results.

1. Introduction

Multi-Person Pose Estimation aims to locate body parts for all persons in an image, such as keypoints on the arms, torsos, and the face. It is a fundamental yet challenging task for many computer vision applications like activity recognition [22] and human re-identification [28]. Achieving accurate localization results, however, is difficult due to the close-interaction scenarios, occlusions and different human scales.

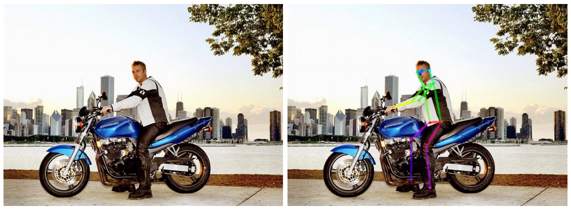

Figure 1. The example of an input image (left) from the COCO test-dev dataset [12] and its estimated pose (right) from our model.

Recently, due to the involvement of deep convolutional neural networks [10, 7], there has been significant progress on the problem of multi-person pose estimation [23, 16, 4, 3, 1, 15, 26]. Existing approaches for multi-person pose estimation can be roughly classified into two frameworks, i.e., top-down framework [23, 16, 4, 3] and bottom-up framework [1, 15, 26]. The former one first detects all human bounding boxes in the image and then estimates the pose within each box independently. The latter one first detects all body keypoints independently and then assembles the detected body joints to form multiple human poses.

Although great progress has been made, it is still an open problem to achieve accurate localization results. First, on the one hand, high-level feature maps with larger receptive fields are required in some challenging cases to infer the invisible and occluded keypoints, e.g., the right knee of the human in Fig. 1. On the other hand, low-level feature maps with larger resolutions are also helpful to the detailed refinement of the keypoints, e.g., the right ankle of the human in Fig. 1. The trade-off between the low-level and high-level feature maps is more complex in real scenarios. Second, the feature fusion is even dynamic and the fused feature maps always remain redundant. Therefore, the information which is more important to the pose estimation should be adap-

[†] Equal contribution.

[*] X. Geng and C. Wang are the corresponding authors.

This work was performed while Kai Su worked as an intern at ByteDance AI Lab.

This research was partially supported by the National Key Research & Development Plan of China (No. 2017YFB1002801), the National Science Foundation of China (61622203), the Collaborative Innovation Center of Novel Software Technology and Industrialization, and the Collaborative Innovation Center of Wireless Communications Technology.

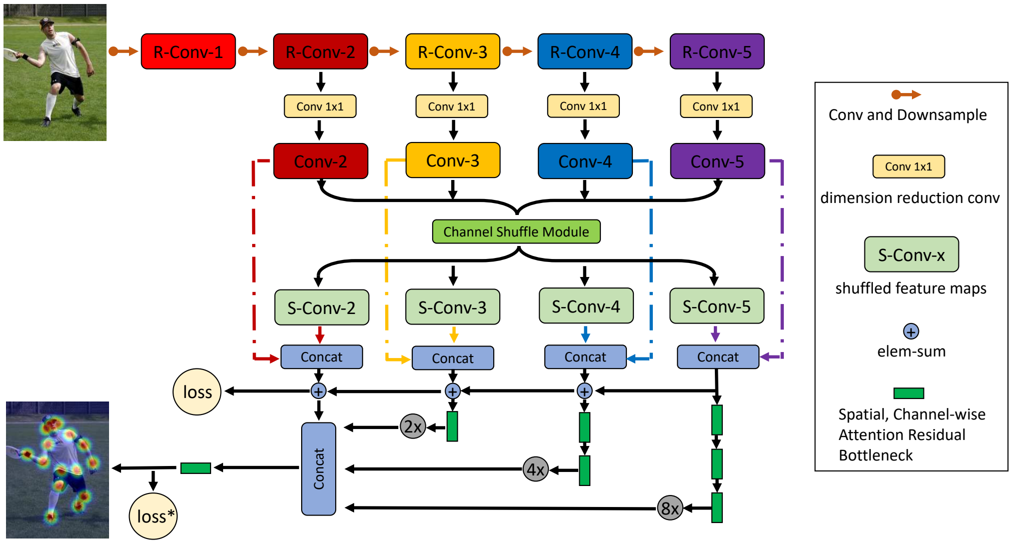

Figure 2. Overview of our architecture. R-Conv-1∼5 are the last residual blocks of different feature maps from the ResNet backbone [7]. R-Conv-2∼5 are first reduced to the same channel dimension of 256 by 1×1 convolution, denoted as Conv-2∼5. S-Conv-2∼5 means the corresponding shuffled feature maps after the Channel Shuffle Module. S-Conv-2∼5 are then concatenated with Conv-2∼5 as the final enhanced pyramid features. Moreover, a Spatial, Channel-wise Attention Residual Bottleneck is proposed to adaptively enhance the fused pyramid feature responses. Loss denotes the L2 loss and loss* means the L2 loss with Online Hard Keypoints Mining [3].

tively highlighted, e.g., with the help of attention mechanism. According to the above analysis, in this paper, we propose a Channel Shuffle Module (CSM) to further enhance the cross-channel communication between the feature maps across all scales. Moreover, a Spatial, Channel-wise Attention Residual Bottleneck (SCARB) is designed to adaptively enhance the fused feature maps both in the spatial and channel-wise context.

To promote the information communication across the channels among the feature maps at different resolution layers, we further exploit the channel shuffle operation proposed in the ShuffleNet [27]. Different from ShuffleNet, in this paper, we creatively adopt the channel shuffle operation to enable the cross-channel information flow among the feature maps across all scales. To the best of our knowledge, the use of the channel shuffle operation to enhance the information of the feature maps is rarely mentioned in previous work for the multi-person pose estimation. As shown in Fig. 2, the proposed Channel Shuffle Module (CSM) performs on the feature maps Conv-2∼5 of different resolutions to obtain the shuffled feature maps S-Conv-2∼5. The idea behind the CSM is that the channel shuffle operation can further recalibrate the interdependencies between the low-level and high-level feature maps.

Moreover, we propose a Spatial, Channel-wise Attention Residual Bottleneck (SCARB), integrating the spatial and channel-wise attention mechanism into the original residual unit [7]. As shown in Fig. 2, by stacking these SCARBs together, we can adaptively enhance the fused pyramid feature responses both in the spatial and channel-wise context. There is a trend of designing networks with attention mechanism, as it is effective in adaptively highlighting the most informative components of an input feature map. However, spatial and channel-wise attention has little been used in the multi-person pose estimation yet.

As one of the classic methods belonging to the top-down framework, Cascaded Pyramid Network (CPN) [3] was the winner of the COCO 2017 keypoint Challenge [13]. Since CPN is an effective structure for the multi-person pose estimation, we apply it as the basic network structure in our experiments to investigate the impact of the enhanced channel-wise and spatial information. We evaluate the two proposed modules on the COCO [12] keypoint benchmark, and ablation studies demonstrate the effectiveness of the Channel Shuffle Module and the Spatial, Channel-wise Attention Residual Bottleneck from various aspects. Experimental results show that our approach achieves the state-of-the-art results.

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

- We propose a Channel Shuffle Module (CSM), which can enhance the cross-channel information communication between the low-level and high-level feature maps.

- We propose a Spatial, Channel-wise Attention Residual Bottleneck (SCARB), which can adaptively enhance the fused pyramid feature responses both in the spatial and channel-wise context.

- Our method achieves the state-of-the-art results on the COCO keypoint benchmark.

The rest of this paper is organized as follows. First, related work is reviewed. Second, our method is described in details. Then ablation studies are performed to measure the effects of different parts of our system, and the experimental results are reported. Finally, conclusions are given.

2. Related Work

This section reviews two aspects related to our method: multi-scale fusion and visual attention mechanism.

2.1. Multi-scale Fusion Mechanism

In the previous work for the multi-person pose estimation, large receptive filed is achieved by a sequential architecture in the Convolutional Pose Machines [23, 1] to implicitly capture the long-range spatial relations among multi-parts, producing the increasingly refined estimations. However, low-level information is ignored along the way. Stacked Hourglass Networks [16, 15] processes the feature maps across all scales to capture various spatial relationships of different resolutions, and adopt the skip layers to preserve spatial information at each resolution. Moreover, the Feature Pyramid Network architecture [11] is integrated in the GlobalNet of the Cascaded Pyramid Network [3], to maintain both the high-level and low-level information from the feature maps of different scales.

2.2. Visual Attention Mechanism

Visual attention has achieved great success in various tasks, such as the network architecture design [8], image caption [2, 25] and pose estimation [4]. SE-Net [8] proposed a "Squeeze-and-Excitation" (SE) block to adaptively highlight the channel-wise feature maps by modeling the channel-wise statistics. However, SE block only considers the channel-wise relationship and ignores the importance of the spatial attention in the feature maps. SCA-CNN [2] proposed Spatial and Channel-wise Attentions in a CNN for image caption. Spatial and channel-wise attention not only encodes where (i.e., spatial attention) but also introduces what (i.e., channel-wise attention) the important visual attention is in the feature maps. However, spatial and channel-wise attention has little been used in the multi-person pose estimation yet. Chu *et al.* [4] proposed the effective multi-context attention model for the human pose estimation. However, our proposed spatial and channel-wise attention residual bottleneck for the multi-person pose estimation has not been mentioned in [4] yet.

3. Method

An overview of our proposed framework is illustrated in Fig. 2. We adopt the effective Cascaded Pyramid Network (CPN) [3] as the basic network structure to explore the effects of the Channel Shuffle Module and the Spatial, Channel-wise Attention Residual Bottleneck for the multi-person pose estimation. We first briefly review the structure of the CPN, and then the detailed descriptions of our proposed modules are presented.

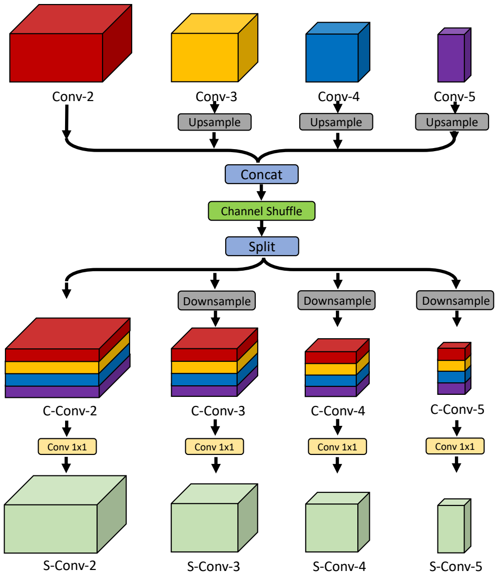

Figure 3. Channel Shuffle Module. The module adopts the channel shuffle operation on the pyramid features Conv-2∼5 to achieve the shuffled pyramid features S-Conv-2∼5 with cross-channel communication between different levels. The groups g is set as 4 here.

3.1. Revisiting Cascaded Pyramid Network

Cascaded Pyramid Network (CPN) [3] is a two-step network structure for the human pose estimation. Given a human box, first, CPN uses the GlobalNet to locate somewhat "simple" keypoints based on the FPN architecture [11]. Second, CPN adopts the RefineNet with the Online Hard Keypoints Mining mechanism to explicitly address the "hard" keypoints.

As shown in Fig. 2, in this paper, for the GlobalNet, the feature maps with different scales (i.e., R-Conv-2∼5) extracted from the ResNet [7] backbone are first reduced to the same channel dimension of 256 by 1×1 convolution, denoted as Conv-2∼5. The proposed Channel Shuffle Module then performs on the Conv-2∼5 to obtain the shuffled feature maps S-Conv-2∼5. Finally, S-Conv-2∼5 are concatenated with the original pyramid features Conv-2∼5 as the final enhanced pyramid features, which will be used as the U-shape FPN architecture. In addition, for the RefineNet, a boosted residual bottleneck with spatial, channel-wise attention mechanism is proposed to adaptively highlight the feature responses transferred from the GlobalNet both in the spatial and channel-wise context.

3.2. CSM: Channel Shuffle Module

As the levels of the feature maps are greatly enriched by the depth of the layers in the deep convolutional neural networks, many visual tasks have made significant improvements, e.g., image classification [7]. However, for the multi-person pose estimation, there are still limitations in the trade-off between the low-level and high-level feature maps. The channel information with different characteristics among different levels can complement and reinforce with each other. Motivated by this, we propose the Channel Shuffle Module (CSM) to further recalibrate the interdependencies between the low-level and high-level feature maps.

As shown in Fig. 3, assuming that the pyramid features extracted from the ResNet backbone are denoted as Conv-2$\sim$5 (with the same channel dimension of 256). Conv-3$\sim$5 are first upsampled to the same resolution as the Conv-2, and then these feature maps are concatenated together. After that, the channel shuffle operation is performed on the concatenated features to fuse the complementary channel information among different levels. The shuffled features are then split and downsampled to the original resolution separately, denoted as C-Conv-2$\sim$5. C-Conv-2$\sim$5 can be viewed as the features consisting of the complementary channel information from feature maps among different levels. After that, we perform 1×1 convolution to further fuse C-Conv-2$\sim$5, and obtain the shuffled features, denoted as S-Conv-2$\sim$5. We then concatenate the shuffled feature maps S-Conv-2$\sim$5 with the original pyramid feature maps Conv-2$\sim$5 to achieve the final enhanced pyramid feature representations. These enhanced pyramid feature maps not only contain the information from the original pyramid features, but also the fused cross-channel information from the shuffled pyramid feature maps.

3.2.1 Channel Shuffle Operation

As described in the ShuffleNet [27], a channel shuffle operation can be modeled as a process composed of "reshape-transpose-reshape" operations. Assuming the concatenated features from different levels as $\boldsymbol{\Psi}$, and the channel dimension of $\boldsymbol{\Psi}$ in this paper is $256 * 4 = 1024$. We first reshape the channel dimension of $\boldsymbol{\Psi}$ into (g, c), where g is the number of groups, $c = 1024/g$. Then, we transpose the channel dimension to (c, g), and flatten it back to 1024. After the channel shuffle operation, $\boldsymbol{\Psi}$ is fully related in the channel context. The number of groups g will be discussed in the ablation studies of the experiments.

3.3. ARB: Attention Residual Bottleneck

Based on the enhanced pyramid feature representations introduced above, we attach our boosted Attention Residual Bottleneck to adaptively enhance the feature responses both in the spatial and channel-wise context. As shown in Fig. 4, our Attention Residual Bottleneck learns the spatial attention weights $\boldsymbol{\beta}$ and the channel-wise attention weights $\boldsymbol{\alpha}$ respectively.

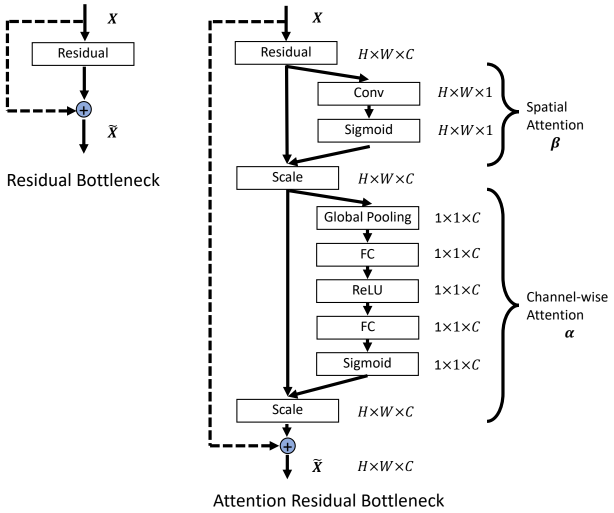

Figure 4. The schema of the original Residual Bottleneck (left) and the Spatial, Channel-wise Attention Residual Bottleneck (right), which is composed of the spatial attention and channel-wise attention. Dashed links indicate the identity mapping.

3.3.1 Spatial Attention

Applying the whole feature maps may lead to sub-optimal results due to the irrelevant regions. Different from paying attention to the whole image region equally, spatial attention mechanism attempts to adaptively highlight the task-related regions in the feature maps.

Assuming the input of the spatial attention is $\boldsymbol{V} \in \mathbb{R}^{H\times W\times C}$, and the output of the spatial attention is $\boldsymbol{V}^{'} \in \mathbb{R}^{H\times W\times C}$, then we can get $\boldsymbol{V}^{'} = \boldsymbol{\beta} * \boldsymbol{V}$, where $*$ means the element-wise multiplication in the spatial context. The spatial-wise attention weights $\boldsymbol{\beta} \in \mathbb{R}^{H\times W}$ is generated by a convolutional operation $\boldsymbol{W} \in \mathbb{R}^{1\times 1\times C}$ followed by a sigmoid function on the input $\boldsymbol{V}$, i.e.,

$$\boldsymbol{\beta} = Sigmoid(\boldsymbol{W}\boldsymbol{V}), \tag{1}$$

where $\boldsymbol{W}$ denotes the convolution weights, and $Sigmoid$ means the sigmoid activation function.

Finally the learned spatial attention weights $\boldsymbol{\beta}$ is rescaled on the input $\boldsymbol{V}$ to achieve the output $\boldsymbol{V}^{'}$.

$$\boldsymbol{v}^{'}_{i,j} = \beta_{i,j} * \boldsymbol{v}_{i,j}, \tag{2}$$

where $*$ means the element-wise multiplication between the i, j-th element of $\boldsymbol{\beta}$ and $\boldsymbol{V}$ in the spatial context.

3.3.2 Channel-wise Attention

Since convolutional filters perform as a pattern detector, and each channel of a feature map after the convolutional operation is the feature activations of the corresponding convolutional filters. The channel-wise attention mechanism can be viewed as a process of adaptively selecting the pattern detectors, which are more important to the task.

Assuming the input of the channel-wise attention is $\boldsymbol{U} \in \mathbb{R}^{H\times W\times C}$, and the output of the channel-wise attention is $\boldsymbol{U}^{'} \in \mathbb{R}^{H\times W\times C}$, then we can get $\boldsymbol{U}^{'} = \boldsymbol{\alpha} * \boldsymbol{U}$, where $*$ means the element-wise multiplication in the channel-wise context, $\boldsymbol{\alpha} \in \mathbb{R}^{C}$ is the channel-wise attention weights. Following the SE-Net [8], channel-wise attention can be modeled as a process consisting of two steps, i.e., squeeze and excitation step respectively.

In the squeeze step, global average pooling operation is first performed on the input $\boldsymbol{U}$ to generate the channel-wise statistics $\boldsymbol{z} \in \mathbb{R}^{C}$, where the c-th element of $\boldsymbol{z}$ is calculated by

$$z_c = \frac{1}{H\times W}\sum_{i=1}^{H}\sum_{j=1}^{W}\boldsymbol{u}_c(i,j), \tag{3}$$

where $\boldsymbol{u}_c \in \mathbb{R}^{H\times W}$ is the c-th element of the input $\boldsymbol{U}$.

In the excitation step, a simple gating mechanism with a sigmoid activation is performed on the channel-wise statistics $\boldsymbol{z}$, i.e.,

$$\boldsymbol{\alpha} = Sigmoid(\boldsymbol{W}_2(\sigma(\boldsymbol{W}_1(\boldsymbol{z})))), \tag{4}$$

where $\boldsymbol{W}_1 \in \mathbb{R}^{C\times C}$ and $\boldsymbol{W}_2 \in \mathbb{R}^{C\times C}$ denotes two fully connected layers, σ means the ReLU activation function [14], and $Sigmoid$ means the sigmoid activation function.

Finally, the learned channel-wise attention weights $\boldsymbol{\alpha}$ is rescaled on the input $\boldsymbol{U}$ to achieve the output of the channel-wise attention $\boldsymbol{U}^{'}$, i.e.,

$$\boldsymbol{u}_c^{'} = \alpha_c * \boldsymbol{u}_c, \tag{5}$$

where $*$ means the element-wise multiplication between the c-th element of $\boldsymbol{\alpha}$ and $\boldsymbol{U}$ in the channel-wise context.

As shown in Fig. 4, assuming the input of the residual bottleneck is $\boldsymbol{X} \in \mathbb{R}^{H\times W\times C}$, the attention mechanism is performed on the non-identity branch of the residual module, and the spatial, channel-wise attention act before the summation with the identity branch. There are two different implementation orders of the spatial attention and channel-wise attention in the residual bottleneck [7], i.e., SCARB: Spatial, Channel-wise Attention Residual Bottleneck and CSARB: Channel-wise, Spatial Attention Residual Bottleneck respectively, which are described as follows.

3.3.3 SCARB: Spatial, Channel-wise Attention Residual Bottleneck

The first type applies the spatial attention before the channel-wise attention, as shown in Fig. 4. All processes are summarized as follows:

$$\begin{aligned}\boldsymbol{X}^{'} &= F(\boldsymbol{X}),\\ \boldsymbol{Y} &= \boldsymbol{\alpha} * (\boldsymbol{\beta} * \boldsymbol{X}^{'}),\\ \widetilde{\boldsymbol{X}} &= \sigma(\boldsymbol{X} + \boldsymbol{Y}),\end{aligned} \tag{6}$$

where the function $F(\boldsymbol{X})$ represents the residual mapping to be learned in the ResNet [7], $\widetilde{\boldsymbol{X}}$ is the output attention feature maps with the enhanced spatial and channel-wise information.

3.3.4 CSARB: Channel-wise, Spatial Attention Residual Bottleneck

Similarly, the second type is a model with channel-wise attention implemented first, i.e.,

$$\begin{aligned}\boldsymbol{X}^{'} &= F(\boldsymbol{X}),\\ \boldsymbol{Y} &= \boldsymbol{\beta} * (\boldsymbol{\alpha} * \boldsymbol{X}^{'}),\\ \widetilde{\boldsymbol{X}} &= \sigma(\boldsymbol{X} + \boldsymbol{Y}).\end{aligned} \tag{7}$$

The choice of the SCARB and CSARB will be discussed in the ablation studies of the experiments.

4. Experiments

Our multi-person pose estimation system follows the top-down pipeline. First, a human detector is applied to generate all human bounding boxes in the image. Then for each human bounding box, we apply our proposed network to predict the corresponding human pose.

4.1. Experimental Setup

4.1.1 Datasets and Evaluation Criterion

We evaluate our model on the challenging COCO keypoint benchmark [12]. Our models are only trained on the COCO trainval dataset (includes $57K$ images and $150K$ person instances) with no extra data involved. Ablation studies are validated on the COCO minival dataset (includes $5K$ images). The final results are reported on the COCO test-dev dataset (includes $20K$ images) compared with the public state-of-the-art results. We use the official evaluation metric [12] that reports the OKS-based AP (average precision) in the experiments, where the OKS (object keypoints similarity) defines the similarity between the predicted pose and the ground truth pose.

Table 1. Ablation study on the Channel Shuffle Module (CSM) with different groups g on the COCO minival dataset. CSM-g denotes the Channel Shuffle Module with g groups. The Attention Residual Bottleneck is not used in this experiment.

Method	AP
CPN (baseline)	69.4
CPN + CSM-2	70.4
CPN + CSM-4	**71.7**
CPN + CSM-8	71.4
CPN + CSM-16	71.2
CPN + CSM-32	70.1
CPN + CSM-64	70.7
CPN + CSM-128	71.0
CPN + CSM-256	71.6

Table 2. Ablation study on the Attention Residual Bottleneck on the COCO minival dataset. SCARB denotes the Spatial, Channel-wise Attention Residual Bottleneck, CSARB denotes the Channel-wise, Spatial Attention Residual Bottleneck. The Channel Shuffle Module is not used in this experiment.

Method	AP
CPN (baseline)	69.4
CPN + CSARB	70.4
CPN + SCARB	**70.8**

Table 3. Component analysis on the Channel Shuffle Module with 4 groups (CSM-4) and the Spatial, Channel-wise Attention Residual Bottleneck (SCARB) on the COCO minival dataset. Based on the baseline CPN [3], we gradually add the CSM-4 and SCARB for ablation studies. The last line shows the total improvement compared with the baseline CPN.

Method	CSM-4	SCARB	AP
CPN (baseline)			69.4
CPN + CSM-4	√		71.7
CPN + SCARB		√	70.8
CPN + CSM-4 + SCARB	√	√	**72.1**

4.1.2 Training Details

Our pose estimation model is implemented in Pytorch [18]. For the training, 4 V100 GPUs on a server are used. Adam [9] optimizer is adpoted. The base learning rate is set to $5e-4$, and is decreased by a factor of 0.1 at 90 and 120 epochs, and finally we train for 140 epochs. The input size of the image for the network is made to a fixed aspect ratio of height : width = $4:3$, e.g., 256×192 is used as the default resolution, the same as the CPN [3]. L2 loss is used for the GlobalNet, and following the CPN, we only punish the top 8 keypoints losses in the Online Hard Keypoint Mining of the RefineNet. Data augmentation during the training includes the random rotation ($-40° \sim +40°$) and the random scale ($0.7 \sim 1.3$).

Our ResNet backbone is initialized with the weights of the public-released Imagenet [20] pre-trained model. ResNet backbones with 50, 101 and 152 layers are experimented. ResNet-50 is used by default, unless otherwise noted.

Table 4. Comparison with the 8-stage Hourglass [16], CPN [3] and Simple Baselines [24] on the COCO minival dataset. Their results are cited from [3, 24]. "*" means the model training with the Online Hard Keypoints Mining.

Method	Backbone	Input Size	AP
8-stage Hourglass	-	256 × 192	66.9
8-stage Hourglass	-	256 × 256	67.1
CPN (baseline)	ResNet-50	256 × 192	68.6
CPN (baseline)	ResNet-50	384 × 288	70.6
CPN* (baseline)	ResNet-50	256 × 192	69.4
CPN* (baseline)	ResNet-50	384 × 288	71.6
Simple Baselines	ResNet-50	256 × 192	70.6
Simple Baselines	ResNet-50	384 × 288	72.2
Ours*	ResNet-50	256 × 192	**72.1**
Ours*	ResNet-50	384 × 288	**73.8**

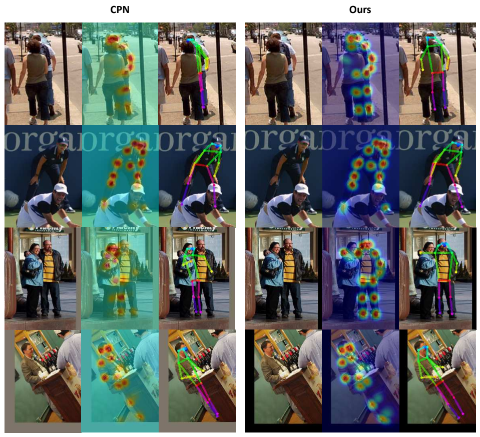

Figure 5. Visual heatmaps of CPN and our model on the COCO minival dataset. From left to right are input images, predicted heatmaps and predicted poses. Best viewed in zoom and color.

4.1.3 Testing Details

For the testing, a top-down pipeline is applied. For the COCO minival dataset, we use the human detection results provided by the CPN [3] for the fair comparison, which reports the human detection AP 55.3. For the COCO test-dev dataset, we adopt the SNIPER [21] as the human detector, which achieves the human detection AP 58.1. Following the common practice in [3, 16], the keypoints are estimated on the averaged heatmaps of the original and flipped image. A quarter offset in the direction from the highest response to the second highest response is used to obtain the final keypoints.

4.2. Component Ablation Studies

In this section, we conduct the ablation studies on the Channel Shuffle Module and the Attention Residual Bottleneck on the COCO minival dataset. The ResNet-50 backbone and the input size of 256×192 are used by default in the all ablation studies.

Table 5. Comparison of final results on the COCO test-dev dataset. **Top:** methods in the literature, trained only with the COCO trainval dataset. **Middle:** results submitted to the COCO test-dev leaderboard [13]. "*" means that the method involves extra data for the training. "+" indicates the results using the ensembled models. **Bottom:** the results of our single model, trained only with the COCO trainval dataset. ⋇ indicates the results using the single model with flip and rotation testing strategy.

Method	Backbone	Input Size	AP	AP .5	AP .75	AP (M)	AP (L)	AR
CMU-Pose [1]	-	-	61.8	84.9	67.5	57.1	68.2	66.5
Mask-RCNN [6]	ResNet-50-FPN	-	63.1	87.3	68.7	57.8	71.4	-
Associative Embedding [15]	-	512 × 512	65.5	86.8	72.3	60.6	72.6	70.2
G-RMI [17]	ResNet-101	353 × 257	64.9	85.5	71.3	62.3	70.0	69.7
CPN [3]	ResNet-Inception	384 × 288	72.1	91.4	80.0	68.7	77.2	78.5
Simple Baselines [24]	ResNet-101	384 × 288	73.2	91.4	80.9	69.7	79.5	78.6
Simple Baselines [24]	ResNet-152	384 × 288	73.8	91.7	81.2	70.3	80.0	79.1
FAIR Mask R-CNN* [13]	ResNet-101-FPN	-	69.2	90.4	77.0	64.9	76.3	75.2
G-RMI* [13]	ResNet-152	353 × 257	71.0	87.9	77.7	69.0	75.2	75.8
oks* [13]	-	-	72.0	90.3	79.7	67.6	78.4	77.1
bangbangren*+ [13]	ResNet-101	-	72.8	89.4	79.6	68.6	80.0	78.7
CPN+ [13]	ResNet-Inception	384 × 288	73.0	91.7	80.9	69.5	78.1	79.0
Ours	ResNet-50	256 × 192	71.4	91.3	79.8	68.3	77.1	77.1
Ours	ResNet-50	384 × 288	73.2	91.9	81.0	69.6	79.3	78.5
Ours	ResNet-101	256 × 192	71.8	91.3	80.1	68.7	77.3	78.8
Ours	ResNet-101	384 × 288	73.8	91.7	81.4	70.4	79.6	80.3
Ours	ResNet-152	256 × 192	72.3	91.4	80.6	69.2	77.8	79.2
Ours	ResNet-152	384 × 288	74.3	91.8	81.9	70.7	80.2	80.5
Ours⋇	ResNet-101	384 × 288	74.1	91.8	81.7	70.6	80.0	80.4
Ours⋇	ResNet-152	384 × 288	**74.6**	**91.8**	**82.1**	**70.9**	**80.6**	**80.7**

4.2.1 Groups g in the Channel Shuffle Module

In this experiment, we explore the performances of the Channel Shuffle Module with different groups on the COCO minival dataset. CSM-g denotes the Channel Shuffle Module with g groups and the groups g controls the degree of the cross-channel feature maps fusion. The ResNet-50 backbone and the input size of 256×192 are used by default, and the Attention Residual Bottleneck is not used here. As the Table 1 shows, 4 groups achieves the best AP of 71.7. It indicates that when only using the Channel Shuffle Module with 4 groups (CSM-4), we can achieve 2.3 AP improvement compared with the baseline CPN. Therefore, 4 groups (CSM-4) is selected finally.

4.2.2 Attention Residual Bottleneck: SCARB and CSARB

In this experiment, we explore the effects of different implementation orders of the spatial attention and the channel-wise attention in the Attention Residual Bottleneck, i.e., SCARB and CSARB. The ResNet-50 backbone and the input size of 256×192 are used by default, and the Channel Shuffle Module is not used here. As shown in Table 2, the SCARB achieves the best AP of 70.8. It indicates that when only using the SCARB, our model outperforms the baseline CPN by 1.4 AP. Therefore, SCARB is selected by default.

4.2.3 Component Analysis

In this experiment, we analyze the importance of each proposed component on the COCO minival dataset, i.e., the Channel Shuffle Module and the Attention Residual Bottleneck. According to Table 1 and 2, the Channel Shuffle Module with 4 groups (CSM-4) and the Spatial, Channel-wise Attention Residual Bottleneck (SCARB) are selected finally. Accroding to Table 3, compared with the 69.4 AP of the baseline CPN, with only the CSM-4 used, we can achieve 71.7 AP, and with only the SCARB used, we can achieve 70.8 AP. With all the proposed components used, we can achieve 72.1 AP, with the improvement of 2.7 AP over the baseline CPN.

4.3. Comparisons on COCO minival dataset

Table 4 compares our model with the 8-stage Hourglass [16], CPN [3] and Simple Baselines [24] on the COCO minival dataset. The human detection AP of the 8-stage Hourglass and CPN are the same 55.3 as ours. The human detection AP reported in the Simple Baselines is 56.4. Compared with the 8-stage Hourglass, both methods use an input size of 256×192, our model has an improvement of 5.2 AP. CPN and our model both use the Online Hard Keypoints Mining, our model outperforms the CPN by 2.7 AP for the input size of 256×192 and 2.2 AP for the input size of 384×288. Compared with the Simple Baselines, our model outperforms 1.5 AP for the input size of 256×192, and 1.6 AP for the input size of 384×288. Fig. 6 demonstrates the visual heatmaps of CPN and our model on the

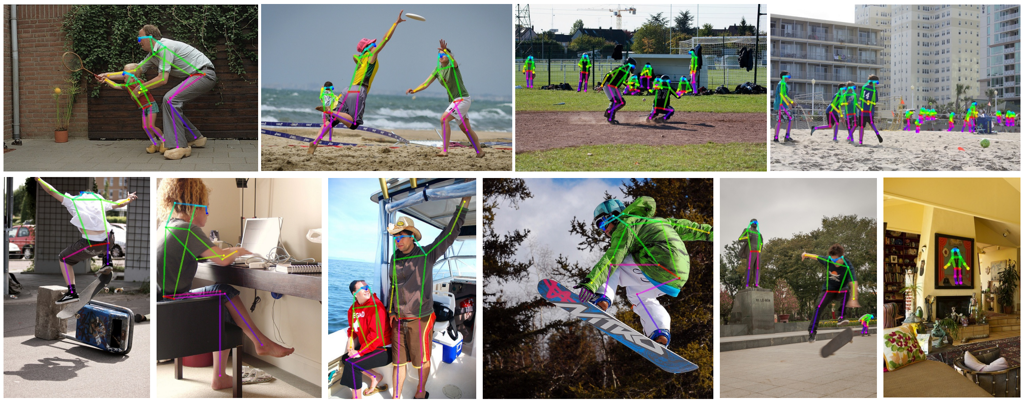

Figure 6. Qualitative results of our model on the COCO test-dev dataset. Our model deals well with the diverse poses, occlusions and cluttered scenes.

Table 6. Comparison between the human detection performance and the pose estimation performance on the COCO test-dev dataset. All pose estimation methods are trained with the ResNet-152 backbone and the 384 × 288 input size.

Pose Method	Det Method	Human AP	Pose AP
Simple Baselines [24]	Faster-RCNN [19]	**60.9**	73.8
Ours	Deformable [5]	45.8	72.9
Ours	- [3]	57.2	73.8
Ours	SNIPER [21]	58.1	**74.3**

COCO minival dataset. As shown in Fig. 6, our model still works in the scenarios (e.g., close-interactions, occlusions) where CPN can not well deal with.

4.4. Experiments on COCO test-dev dataset

In this section, we compare our model with the state-of-the-art methods on the COCO test-dev dataset, and analyze the relationships between the human detection performance and the corresponding pose estimation performance.

4.4.1 Comparison with the state-of-the-art Methods

Table 5 compares our model with other state-of-the-art methods on the COCO test-dev dataset. For the CPN, a human detector with human detection AP 62.9 on the COCO minival dataset is used. For the Simple Baselines, a human detector with human detection AP 60.9 on the COCO test-dev dataset is used. Without extra data for training, our single model can achieve 73.8 AP with the ResNet-101 backbone, and 74.3 AP with the ResNet-152 backbone, which outperform both CPN's single model 72.1 AP, ensembled model 73.0 AP and Simple Baselines 73.8 AP. Moreover, when using the averaged heatmaps of the original, flipped and rotated ($+30°$, $-30°$ is used here) images, our single model can achieve 74.1 AP with the ResNet-101 backbone, and 74.6 AP with the ResNet-152 backbone. Fig. 6 demonstrates the poses predicted by our model on the COCO test-dev dataset.

4.4.2 Human Detection Performance

Table 6 shows the relationships between the human detection performance and the corresponding pose estimation performance on the COCO test-dev dataset. Our model and Simple Baselines [24] are compared in this experiment. Both models are trained with the ResNet-152 backbone and the 384 × 288 input size. The Simple Baselines adopts the Faster-RCNN [19] as the human detector, which reports the human detection AP 60.9 in their paper. For our model, we adopt the SNIPER [21] as the human detector, which achieves the human detection AP 58.1. Moreover, we also use the Deformable Convolutional Networks [5] (achieves the human detection AP 45.8) and the human detection results provided by the CPN [3] (reports the human detection AP 57.2) for comparison.

From the table, we can see that the pose estimation AP gains increasingly when the human detection AP increases. For example, when the human detection AP increases from 57.2 to 58.1, the pose estimation AP of our model increases from 73.8 to 74.3. However, although the human detection AP 60.9 of the Simple Baselines is higher than ours 58.1 AP, the pose estimation AP 73.8 of the Simple Baselines is lower than ours 74.3 AP. Therefore, we can conclude that it is more important to enhance the accuracy of the pose estimator than the human detector.

5. Conclusions

In this paper, we tackle the multi-person pose estimation with the top-down pipeline. The Channel Shuffle Module (CSM) is proposed to promote the cross-channel information communication among the feature maps across all scales, and a Spatial, Channel-wise Attention Residual Bottleneck (SCARB) is designed to adaptively highlight the fused pyramid feature maps both in the spatial and channel-wise context. Overall, our model achieves the state-of-the-art performance on the COCO keypoint benchmark.

References

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

[2] L. Chen, H. Zhang, J. Xiao, L. Nie, J. Shao, W. Liu, and T.-S. Chua. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5659–5667, 2017.

[3] Y. Chen, Z. Wang, Y. Peng, Z. Zhang, G. Yu, and J. Sun. Cascaded pyramid network for multi-person pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7103–7112, 2018.

[4] X. Chu, W. Yang, W. Ouyang, C. Ma, A. L. Yuille, and X. Wang. Multi-context attention for human pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1831–1840, 2017.

[5] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 764–773, 2017.

[6] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pages 2980–2988. IEEE, 2017.

[7] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[8] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

[9] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[10] 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.

[11] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In *CVPR*, volume 1, page 4, 2017.

[12] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[13] MS-COCO. Coco keypoint leaderboard. `http://cocodataset.org/`.

[14] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010.

[15] A. Newell, Z. Huang, and J. Deng. Associative embedding: End-to-end learning for joint detection and grouping. In *Advances in Neural Information Processing Systems*, pages 2274–2284, 2017.

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

[17] G. Papandreou, T. Zhu, N. Kanazawa, A. Toshev, J. Tompson, C. Bregler, and K. Murphy. Towards accurate multi-person pose estimation in the wild. In *CVPR*, volume 3, page 6, 2017.

[18] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. 2017.

[19] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[20] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[21] B. Singh, M. Najibi, and L. S. Davis. Sniper: Efficient multi-scale training. In *Advances in Neural Information Processing Systems*, pages 9333–9343, 2018.

[22] C. Wang, Y. Wang, and A. L. Yuille. An approach to pose-based action recognition. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 915–922. IEEE, 2013.

[23] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. Convolutional pose machines. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4724–4732, 2016.

[24] B. Xiao, H. Wu, and Y. Wei. Simple baselines for human pose estimation and tracking. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 466–481, 2018.

[25] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo. Image captioning with semantic attention. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4651–4659, 2016.

[26] A. Zanfir, E. Marinoiu, M. Zanfir, A.-I. Popa, and C. Sminchisescu. Deep network for the integrated 3d sensing of multiple people in natural images. In *Advances in Neural Information Processing Systems*, pages 8420–8429, 2018.

[27] X. Zhang, X. Zhou, M. Lin, and J. Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6848–6856, 2018.

[28] L. Zheng, Y. Huang, H. Lu, and Y. Yang. Pose invariant embedding for deep person re-identification. *arXiv preprint arXiv:1701.07732*, 2017.