IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 20 July 2023, accepted 24 August 2023, date of publication 31 August 2023, date of current version 8 September 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3310817

APPLIED RESEARCH

EDite-HRNet: Enhanced Dynamic Lightweight High-Resolution Network for Human Pose Estimation

LIYUHENG RUI[1], **YANYAN GAO**[2], **AND HAOPAN REN**[2]
[1]School of Physics, Beihang University, Beijing 100191, China
[2]School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China

Corresponding author: Liyuheng Rui (2495088137@qq.com)

This work was supported by the National Natural Science Foundation of China under Grant 11205012.

ABSTRACT Lightweight pose estimation models have been widely used in devices with different computing powers, providing convenience for numerous downstream tasks, such as gait estimation, behavior analysis, motion capture, etc. Although these lightweight methods can run on low-performance equipment, their estimation accuracy is low, which seriously affects the actual experience. In order to improve the prediction accuracy of the lightweight human pose estimation methods, we propose an Enhanced Dynamic Lightweight High-Resolution Network (EDite-HRNet) for human pose estimation. Specifically, we propose an Enhanced Dynamic Multi-scale Context (EDMC) block which enhances the features of the simple branch with multi-level features of the complex branch to realize multi-level features fusion. Moreover, inspired by GhostNet V2, we redesign the Enhanced Dynamic Global Context (EDGC) and the Enhanced Dynamic Multi-scale Context (EDMC) block by adopting GhostNet V2 module with DFC attention to replace ConvBN block in the original blocks. The experimental results on the two datasets (66.1% on the COCO2017 dataset and 86.8% on the MPII dataset), demonstrate that our network achieves the state-of-the-art performance with a slight increase in model complexity.

INDEX TERMS Human pose estimation, lightweight network, computational cost, cross-block feature fusion, long-range dependencies.

I. INTRODUCTION

Lightweight human pose estimation methods has a good balance between the accuracy and calculation of the model, and is more friendly to some edge devices with limited computing power. Therefore, these lightweight pose estimation methods have broader application scenarios, such as vehicle equipment to capture pedestrian motion trajectory, mobile devices to analyze human behavior, etc.

Traditional classical human pose estimation network models [1], [2], [3], [4], [5], [6], [7], [8], [9] have high prediction accuracy, but their parameter and calculation amount are often extremely large. The latest work [10] introduce a self-correctable and adaptable inference (SCAI) method to improve the accuracy on COCO test-dev to 80.6AP. However, it adds computational cost. These methods is difficult to apply in edge devices, and even if it can, it will introduce a huge computing load. Although lightweight pose estimation methods have small calculations and parameters, their accuracy is very low, and it is difficult to meet the actual needs. Simple Baseline [2] uses deconvolution layers to reduce the number of parameters and the computation complexity. Lite-HRNet [11] introduce a lightweight unit, conditional channel weighting, to replace costly pointwise (1 × 1) convolutions in shuffle blocks. Dite-HRNet [12] applies dynamic split convolution and adaptive context modeling to model long-range spatial dependencies for human pose estimation with low model complexity. The lightweight pose estimation method requires less hardware resources and small amount of computation, making it suitable for deployment on different types of edge devices. Generally, how to improve the prediction accuracy of lightweight

The associate editor coordinating the review of this manuscript and approving it for publication was Binit Lukose.

pose estimation method has become the focus of current research.

Existing lightweight networks are mainly inspired from lightweight classification networks. Therefore, we need to learn from the design ideas of lightweight network models to improve the prediction accuracy of lightweight pose estimation models under the condition of small computing loads. MobileNet [13], [14] and ShuffleNet [15], [16] apply separable convolution and group convolution to replace conventional convolution operation. GhostNet adopts cheap operation to maintain similar prediction accuracy, which greatly reduces the computational load of the model. Inspired by these lightweight model methods, our lightweight pose estimation method adopts similar principles.

In order to improve the prediction accuracy of the lightweight pose estimation model, we propose the method (EDite-HRNet). Firstly, we redesign Dynamic Multi-scale Context (DMC) block in Dite-HRNet and propose an Enhanced Dynamic Multi-scale Context (EDMC) block. The EDMC block firstly fuses the hierarchical features of different depths in the complex branch and then aggregates the multi-level features in the simple branch, which can fully learn the abstract features at different layers and the complex cross-block interactions. Compared with the original DMC block, the EDMC block effectively enhances the capability to extract channel information. Secondly, inspired by GhostNet V2 [17], we introduce Ghostnet V2 module with DFC attention to replace ConvBN block in the original Dynamic Global Context (DGC) block and Dynamic Multi-scale Context (DMC) block deployed in Dite-HRNet [12] and generate the Enhanced Dynamic Global Context (EDGC) block. DFC attention consists of horizontal fully connected (FC) layers and vertical FC layers, which involve pixels in a long-range along their respective directions, and produce a global attention map. As a result, the EDGC block and EDMC block are able to capture long-range spatial information and represent global features.

Our main contributions include:

We propose an Enhanced Dynamic Multi-scale Context (EDMC) block. The proposed block fuses the hierarchical features of different depths in the complex branch and then aggregates the multi-level features in the simple branch, which can fully learn the abstract features at different layers and the complex cross-block interactions. By introducing Ghostnet V2 module with DFC attention into DMC and DGC block of EDite-HRNet, we propose the EDGC and EDMC block to expand the receptive field in a cost-effective fashion and enhance the modeling performance for long-range dependencies. By designing EDMC and EDGC blocks, we propose a novel lightweight human pose estimation method (EDite-HRNet), which achieves state-of-the-art (SOTA) performance with a slight increase in model complexity.

II. RELATED WORK

A. LIGHTWEIGHT HUMAN POSE ESTIMATION

Efficiencies of human pose estimation networks have drawn wide attention recently. Small HRNet [11] reduces the width and depth of the original HRNet [8]. Lite-HRNet [11] replaces each residual block in the Small HRNet with a conditional channel weighting block by adopting channel attention mechanism [18]. Dite-HRNet [12] adapts dynamic multi-scale context block and dynamic global context block by using Dynamic Split Convolution and Adaptive Context Modeling
mechanisms.

B. EFFICIENT CNN BLOCKS

Efficient CNN blocks are the core design of lightweight architectures. MobileNet [13] adapts depth-wise separable convolutions to efficiently trades off between latency and accuracy. MobileNet V2 [19] adapts inverted residuals and linear bottlenecks. The architecture of MobileNet V3 [14] is built from a network architecture search algorithm called NetAdpt. ShuffleNet [15] primarily adapts pointwise group convolution operations and channel shuffle operations. ShuffleNet V2 [16] considers memory access cost and platform characteristics additionally, and derives several practical guidelines for efficient network design. GhostNet [20] adapts a novel Ghost module with a series of linear transformations. GhostNet V2 [17] enhances the Ghost module with a novel decoupled fully connected attention mechanism.

C. SPATIAL DEPENDENCY MODELING

Deeply stacked convolution layers capture long-range spatial dependency, which riches the global understanding of spatial information in a large field of view. However, they are not computationally efficient. Non-local network [21] adopts a self-attention mechanism to model pixel-wise spatial relations in a single layer. Global context network [22] simplifies the non-local network to a lightweight structure, causing almost no performance degradation.

D. DYNAMIC CNN ARCHITECTURES

Dynamic architectures exploit more efficient feature representations. PP-LCNet [23] dynamically changes operation branches based on the sizes and positions of model inputs. Dynamic convolution [24] and CondConv [25] mix different convolution kernels by generating weights through the attention mechanism.

III. METHOD

A. DITE-HRNet NETWORK AND LIGHTWEIGHT BLOCKS

As shown in Figure 1, the EDite-HRNet network model consists of four stages, starting from the first stage to gradually add low-resolution branches, and finally adding four parallel multi-resolution branches in the fourth stage.

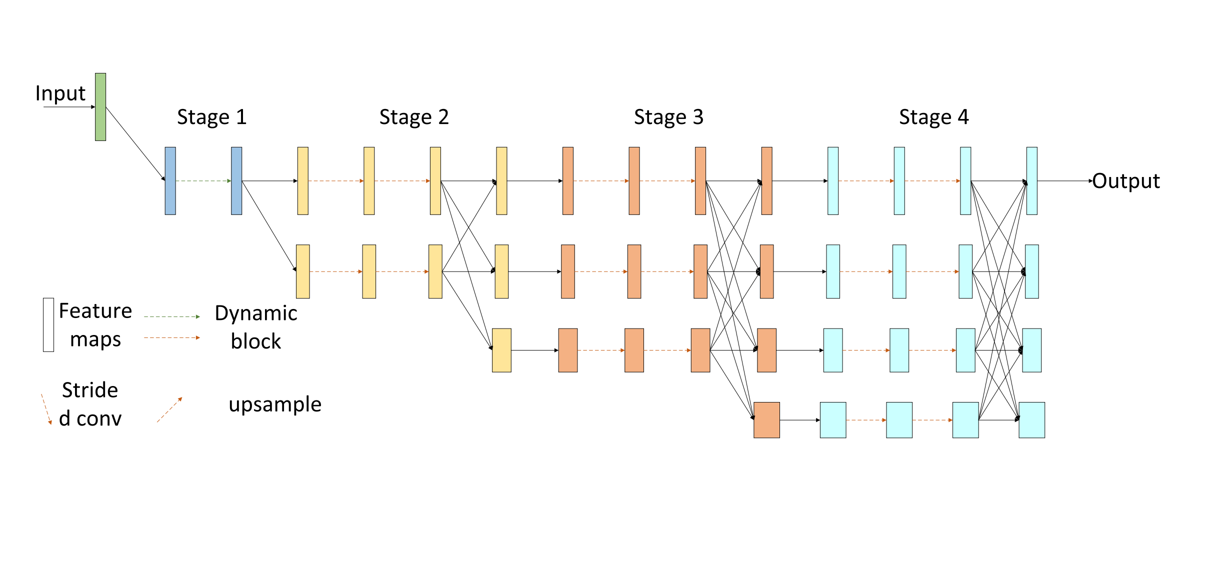

FIGURE 1. **Overall architecture of EDite-HRNet. It adopts the EDite-HRNet network structure as the backbone and adds new EDGC and EDMC blocks. The same color modules in the figure represent the same stage, and the feature maps of different sizes in the figure are reduced by half from top to bottom.**

A low-resolution branch is added at each new stage, and the image resolution of the new low-resolution branch is reduced by half compared with the one of the previous branch, but the number of channels is doubled. EDite-HRNet adopts the Dite-HRNet network as the backbone network.

To obtain a more efficient lightweight pose estimation network, DiteHRNet [12] designed two lightweight blocks, Dynamic Global Context (DGC) block and Dynamic Multi-scale Context (DMC) block. Although the two lightweight blocks in Dite-HRNet can greatly reduce the calculation load of the model, it also leads to a significant reduction in the accuracy of the model. To improve the accuracy of lightweight blocks, we redesigned these lightweight blocks.

B. ENHANCED DMC BLOCK AND DGC BLOCK

As a versatile backbone network, HRNet [8] has been widely used in various types of vision tasks, such as pose estimation, target detection and semantic segmentation. Therefore, there are many works followed HRNet to improve performance, such as Small HRNet [6], Lite-HRNet [11], Dite-HRNet [12], etc. Correspondingly, considering the performance of HRNet, the same backbone network structure is also adopted in our work. More precisely, we design an enhancement method (EDite-HRNet) based on Dite-HRNet, which is a lightweight version of HRNet.

When investigating the design of the Dite-HRNet network structure, we have considered how to achieve a balance between prediction accuracy and computing cost. Although Dite-HRNet becomes the most lightweight model, its accuracy drops significantly. We mainly identifies two main problems of Dite-HRNet. The first one is the lightweight design of the DMC block. The DMC block splits the input channel into two branches. One branch performs a series of convolution operations, while the other keeps the other branch unchanged. These two branches are connected to each other at the end of the block. The two branches in the same block have an asymmetric structure, that is, their convolutional layer structures and numbers are different. Although the model becomes lightweight by applying this approach, the features in the branch without convolutional operations are not extracted, resulting in a decrease in the accuracy. The second one is that the group convolution is performed directly after applying the DKA block, which reduces the long-distance dependency of each group of features.

To conquer the two limitations mentioned above, we propose an Enhanced DMC block and an Enhanced DGC block, as shown in Figure 2. Different from the original asymmetric structure, our novel asymmetric structure incorporates the fusion of multi-scale features, enhancing the ability to capture long-distance dependent features. We apply the design idea of g-GhostNet [26] to the DMC block, which maintains the features of each sub-block in the DMC block, fuses with the features of the simple branch that did not perform a series of convolutions, and finally concatenates the features of two branches according to the dimension. This method can make full use of the characteristics of the branches with and without convolutional operations, and improve the overall accuracy of the model. In addition, in order to learn the long-distance dependency of each set of features, we apply the design idea of GhostNet V2 [17] to the DMC and DGC blocks. We replace the group convolution layer with the ghost module in GhostNet V2, which effectively learns the key feature information.

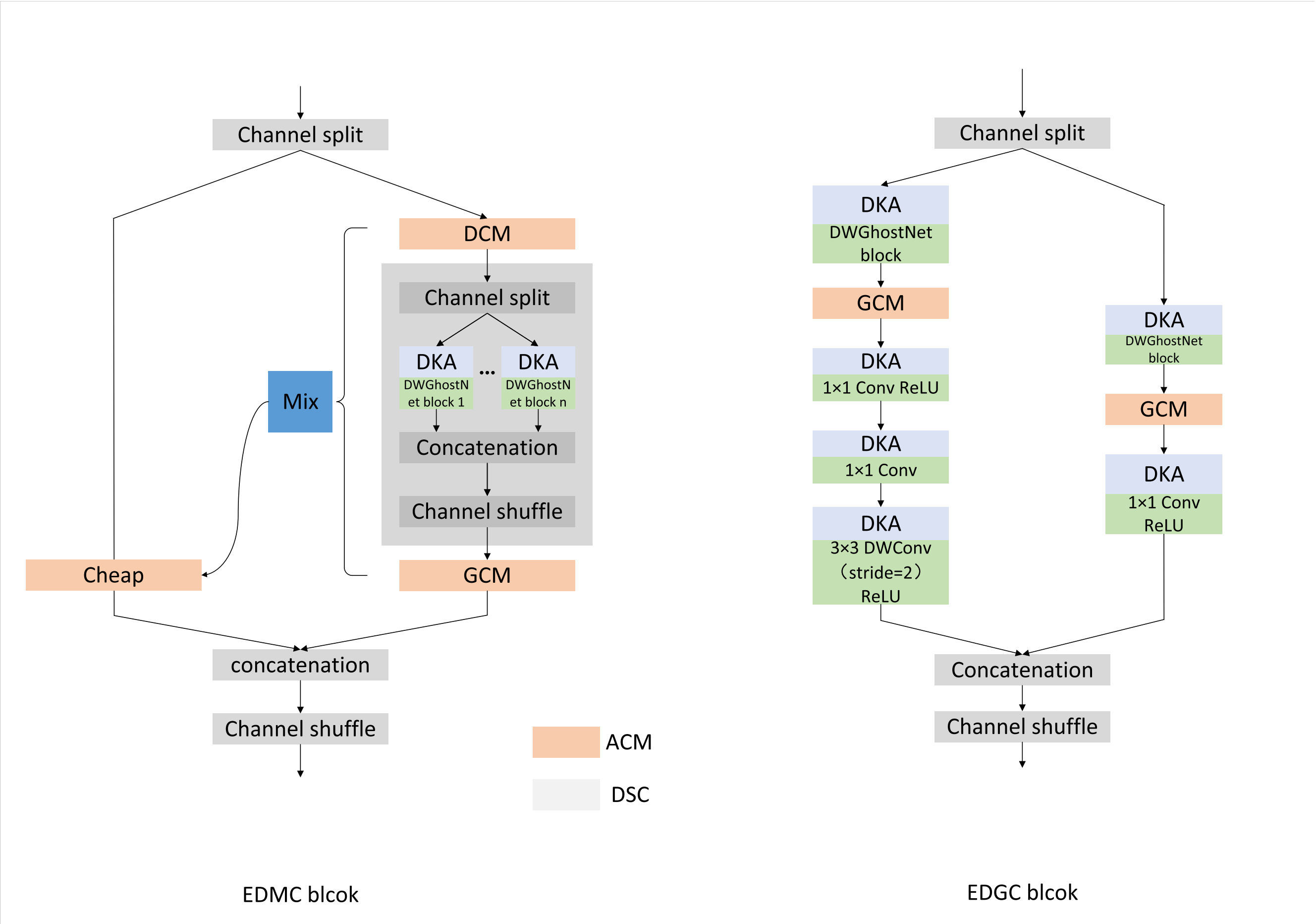

FIGURE 2. The structure of enhanced dynamic multis-cale context (EDMC) block and enhanced dynamic global context (EDGC) block. In the EDMC block, the feature information of different dimensions is stitched together through the Mix operation, and then the information is introduced into another branch through the cheap operation to experiment with the fusion of feature information.

In the next subsection, we will describe our design of two blocks in detail: the EDMC formulaic representation and the Enhanced ConvBN block in EDGC block and in EDMC block.

C. EDMC FORMULAIC REPRESENTATION

To make the network structure more lightweight, a two-branch structure is adopted in the DMC block. One branch does not perform any convolutional operation at all, while the other branch performs convolutional operations for multiple times. Features from these two branches are concatenated and shuffled along the channels. Although such design can reduce the calculation cost of the model, since a branch does not perform convolution operations, the representation capacity of the features is limited, which further limits the overall performance of the model. Inspired by g-GhostNet [26], we design a multi-stage fusion operation, which can effectively improves the overall accuracy of the model.

The DMC block mainly consists of three blocks and two operations, which are the DCM block, the DSC block and the GCM block, as well as the concatenation and the channel shuffle operation. The DCM block is mainly for modeling spatial context relations of the branches with different resolutions densely in a single stage. The DSC block is for extracting spatial information through multiple convolution kernels of different sizes, and integrating them together through a single convolution layer. The GCM block is mainly for modeling the global spatial dependencies of each branch in the network separately. Finally, the features of the two branches are fused by concatenation and channel shuffle operations as the output of the DMC block.

The process of the original DMC module can be formulated as:

$$\begin{aligned} Y_1, Y_2 &= L_s(X) \\ Y_2 &= B_3(B_2(B_1(Y_2))) \\ Output &= L_s(L_c(Y_1 + Y_2)) \end{aligned} \tag{1}$$

where Y_1 and Y_2 denote the feature of the two branches after dividing the original input tensor Y. B_1, B_2, and B_3 denotes DCM, DSC, and GCM, respectively. *Output* denotes the output of the DMC block.

Although the DMC block achieves better accuracy and efficiency, we argues that the feature extraction is incomplete based on the structure, resulting in relatively low accuracy of the model. In g-GhostNet [26], these features are divided into two types, namely complex features and ghost features. Complex features need to be extracted through a series of blocks, while simple features are obtained from shallow features. Inspired by g-GhostNet, we consider features of two branches in the DMC block as complex features and ghost features, respectively. Complex features need to be obtained through a series of convolution blocks. Simple features need to be obtained through the integration of intrinsic features, rather than simply identity mapping.

In the Enhanced DMC (EDMC) block (as shown in figure 3), the input features are denoted as $X \in R^{c \times h \times w}$, where c, h, w denotes the number of channnels, the height and the width of the features, respectively. Complex features are denoted as $Y_n^c \in R^{\lambda \times c \times h \times w}$, and simple features are denoted as $Y_n^s \in R^{(1-\lambda) \times c \times h \times w}$, where λ denotes the ratio of simple features.

Firstly, AEDMC block divides the input features according to the number of channels:

$$Y_1, Y_2 = L_s(X) \tag{2}$$

Complex features are processed by three blocks:

$$Y_2' = B_3'(B_2'(B_1'(Y_2))) \tag{3}$$

By concatenating the feature dimensions of the complex feature branches, EDMC block obtains the concatenated output features of the complex branches:

$$Y_2 = [Y_2^1 + Y_2^2 + Y_2^3] \tag{4}$$

where the subscript denotes the branch index and the superscript denotes the block index.

The simple features of the other branch are obtained by cheap operation:

$$Y_1' = C(Y_1) \tag{5}$$

where the cheap operation C can be simply a 1×1 convolution.

We get a simple feature and a complex feature from two branches. However, the simple feature is not extracted enough. Therefore, to improve the representation capacity, the simple feature is process through a mix operation. From the complex branch, we get the intermediate features $Y_2 \in R^{c' \times h \times w} = [Y_2^1 + Y_2^2 + Y_2^3]$ where c' denotes the total number of channels. The intermediate feature Y_2 obtained from multiple convolution blocks provides sufficient semantic information supplementation for the simple branch. As shown in Figure 3, we first concatenate the outputs of multiple blocks in the channel dimension, and then perform a mix operation on the concatenated features:

$$Y_1' = Y_1' + B_{mix}(Y_2) \tag{6}$$

where B_{mix} denotes the transformation function.

Different from the g-GhostNet method, we removed the average pooling layer, and used a 3×3 convolution kernel instead of a 1×1 convolution kernel, effectively extracting the features of each stage.

D. ENHANCED ConvBN BLOCK IN EDGC BLOCK AND IN EDMC BLOCK

To capture long-distance dependency information, we replace the original convolutional layers with the ghost module from GhostNet v2 [17]. Compared with GhostNet V1 [20], GhostNet V2 uses Decoupled Fully Connected (DFC) attention to enhance the long-distance dependence information of the output of ghost module in different spatial pixels. We replace the convolutional layer in the ConvBN block with the module in GhostNet V2 as shown in Figure 4. Considering the memory limitation of the model and the estimation accuracy, we keep the convolutional layers in other blocks unchanged. Both the ConvBN block in the EDMC and EDGC blocks have been improved, as shown in the red box in Figure 2.

Compared with other self-attention mechanisms, the FC layer with fixed weights has a simple structure, is easy to deploy, and can generate an attention map of the global receptive field. For CNN, feature maps are usually of low rank, so it is unnecessary to densely connect inputs to outputs at different spatial locations. In the DFC attention mechanism, the attention map is decomposed into two FC layers along the vertical and horizontal directions.

The DFC attention mechanism can be formulated as:

$$\begin{aligned} a'_{hw} &= \sum_{h'=1}^{H} F^{H}_{h,h'w} \bigodot X_{h'w}, \quad h=1,2,\ldots,H,\ w=1,2,\ldots,W, \\ a_{hw} &= \sum_{w'=1}^{W} F^{W}_{w,hw'} \bigodot a'_{hw'}, \quad h=1,2,\ldots,H,\ w=1,2,\ldots,W, \end{aligned} \tag{7}$$

where F^H and F^W are transformation weights.

Given the input features $X \in R^{H \times W \times C}$, it can be considered as HW tokens $x_i \in R^C, i = 1, \ldots, HW$. We adopt the DFC attention mechanism to capture long-distance dependency from horizontal and vertical directions. At the same time, due to the transformation in these two directions, the computational complexity of the attention mechanism is also reduced to a certain extent.

In the ghost module from GhostNet V2 [17], the input feature map $X \in R^{H \times W \times C}$ are sent to two branches. One branch is the ghost module for generating the output feature maps, and the other is the DFC module for generating the attention map A. In the typical self-attention, linear transformation are used to transform input feature into query and key for calculating attention maps. The final output $O \in R^{H \times W \times C}$ in the ghost module is the integrated output of the two branches:

$$O = Sigmoid(A) \odot v(X) \tag{8}$$

where $\odot$ denotes the element-wise multiplication. *Sigmoid* is a scaling function to normalize the attention map A into range (0, 1).

The structure of DFC is shown in the figure 5. In the sub-attention mechanism branch, DFC downsamples the input feature, performs horizontal FC and vertical FC, and upsamples the output in the end. Since directly applying of the DFC

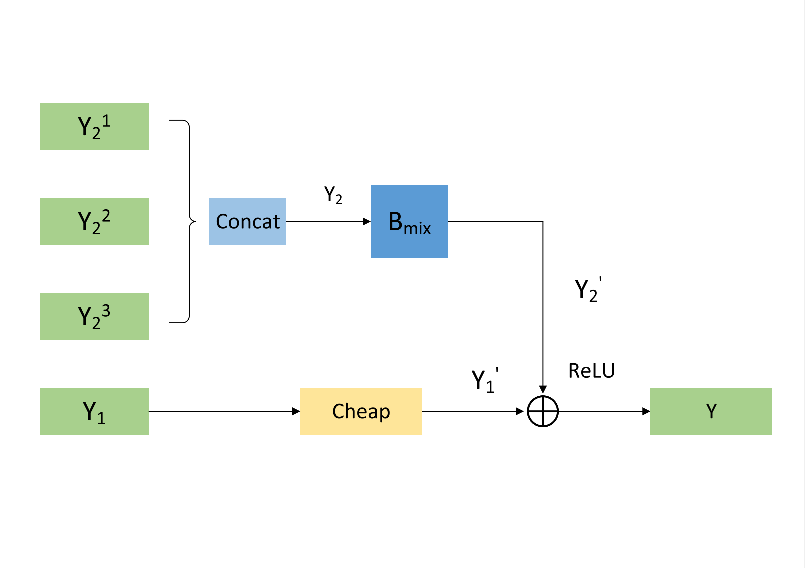

FIGURE 3. The calculation process of the Mix operation. The Mix operation mainly performs feature concatenating and fusion of an original input information and information of three different scales, and can obtain richer multi-scale information.

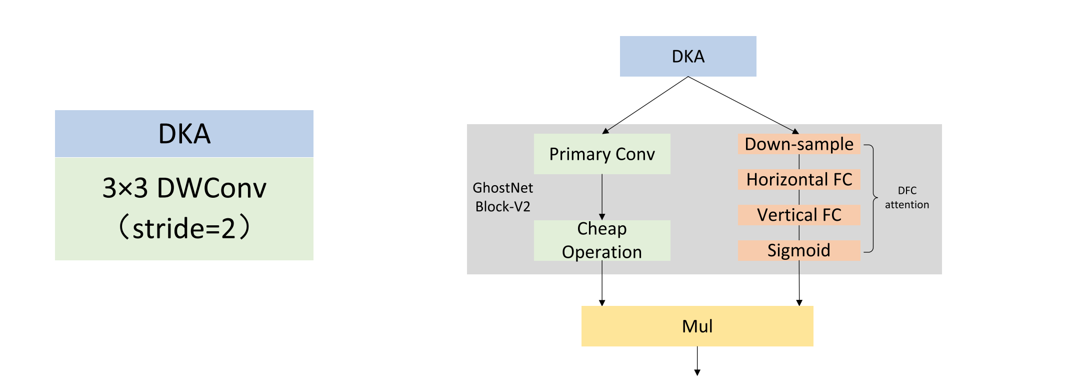

FIGURE 4. The structure of ConvBN. We introduced the Ghostnet v2 module in the ConvBN module, which can model long-distance dependency information.

attention mechanism will increase calculation complexity, the feature map is firstly downsampled horizontally and vertically. The final upsampling is for matching the scale of the input feature map. In the ghost module branch, the intrinsic feature map and the ghost feature map are mainly generated, and finally these two types of feature maps are concatenated. These two parallel branches extract feature maps from mainly two different perspectives.

IV. EXPERIMENTS

A. IMPLEMENTATION DETAILS

1) DATASETS AND EVALUATION METRICS

The COCO2017 dataset contains over 200K images and 250K human instances with a label of 17 keypoints. We experimentally train on train2017 (consists of 57K images and 150K person instances) and evaluate on val2017 (consists of 5K images) and test-dev2017 (consists of 20K images). We adopte the Average Precision (AP) and Average Recall (AR) based on the Object Keypoint Similarity (OKS) as the evaluation criteria of the model. We also conduct experimental tests on the MPII dataset, which contains 25K images and 40K human instances, and also adopt the head-normalized Probability of Correct Keypoint (PCKh) score to evaluate the model accuracy. The crowdpose dataset is a dense human pose estimation dataset containing 20k images and 80k pedestrians. It designs different categories according to different population densities, which improves the generalization ability of the model. It labels every image, every inference point in the box, and the number of key points for each human body is 14. It uses the Average IOU to represent the average IOU of the pedestrian boxes.

2) TRAINING

We conduct experiments on a single GeForce RTX 3080 GPU. The batch size of the experiment mainly depends on the graphics card memory, and it should be as full as possible. All experiments adopt the Adam optimizer with a basic learning rate $2e^{-3}$. In terms of COCO2017 dataset preprocessing, the

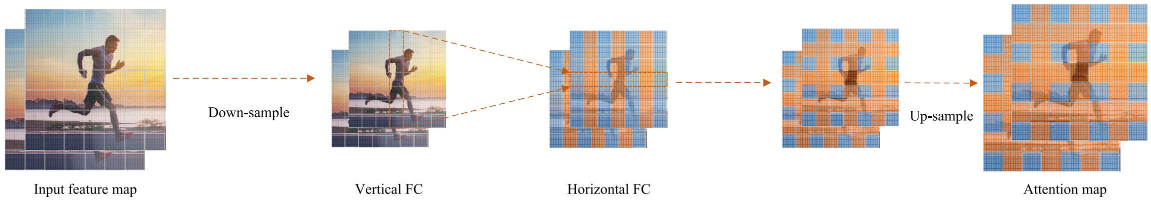

FIGURE 5. The structure of DFC block. The DFC block first down-samples the image to reduce computational complexity, then uses horizontal and vertical convolution to extract features, and finally up-samples to restore image resolution.

TABLE 1. Comparasions of results on the COCO val 2017 set. Pretrain indicates that the backbone network is pre-trained on ImageNet classification task. The image size of the experimental test adopts two commonly used 256×192 and 384×288. The basic unit of parameter quantity is M, and the calculation quantity unit is G.

Large networks											
model	backbone	pretrain	input size	#Params(M)	GFLOPs	AP	AP^{50}	AP^{75}	AP^{M}	AP^{L}	AR
8-stages Hourglass [1]	8-stages Hourglass	N	256 × 192	25.1	14.3	66.9	-	-	-	-	-
CPN [27]	ResNet-50	Y	256 × 192	27.0	6.2	68.6	-	-	-	-	-
Simple Baseline [2]	ResNet-50	Y	256 × 192	34.0	8.9	70.4	88.6	78.3	67.1	77.2	76.3
HRNet [8]	HRNet-W32	Y	256 × 192	28.5	7.10	74.4	90.5	81.9	70.8	81.0	79.8
UDP [28]	HRNet-W32	Y	256 × 192	28.7	7.1	75.2	92.4	82.9	72.0	80.8	80.4
DARK [29]	HRNet-W32	Y	256 × 192	28.5	7.1	75.6	90.5	82.1	71.8	82.8	80.8
Small networks											
DY-Mobile NetV2 1× [19]	DY-MobileNet V2	Y	256 × 192	16.1	1.0	68.2	88.4	76.0	65.0	74.7	74.2
DY-ReLU 1× [30]	MobileNet V2	Y	256 × 192	9.0	1.0	68.1	88.5	76.2	64.8	74.3	-
Mobile NetV2 1× [19]	MobileNet V2	Y	256 × 192	9.6	1.4	64.6	87.4	72.3	61.1	71.2	70.7
Shuffle Net V2 1× [16]	ShuffleNet V2	Y	256 × 192	7.6	1.2	59.9	85.4	66.3	56.6	66.2	66.4
Small HRNet [6]	HRNet-W18	N	256 × 192	1.3	0.5	55.2	83.7	62.4	52.3	61.0	62.1
Lite-HRNet [11]	Lite-HRNet-18	N	256 × 192	1.1	0.2	64.8	86.7	73.0	62.1	70.5	71.2
	Lite-HRNet-30	N	256 × 192	1.8	0.3	67.2	88.0	75.0	64.3	73.1	73.3
Dite-HRNet* [12]	Dite-HRNet-18	N	256 × 192	1.1	0.2	65.2	86.5	73.2	62.6	71.0	71.5
	Dite-HRNet-30	N	256 × 192	1.8	0.3	68.3	88.2	76.2	65.5	74.1	74.2
EDite-HRNet-S	Dite-HRNet-18	N	256 × 192	1.3	0.2	66.1	87.2	73.9	63.4	71.8	72.3
EDite-HRNet-L	Dite-HRNet-18	N	256 × 192	8.8	1.8	70.6	88.6	78.5	67.8	76.7	76.4
Mobile NetV2 1× [19]	MobileNet V2	Y	384 × 288	9.6	3.3	67.3	87.9	74.3	62.8	74.7	72.9
Shuffle Net V2 1× [16]	ShuffleNet V2	Y	384 × 288	7.6	2.8	63.6	86.5	70.5	59.5	70.7	69.7
Small HRNet [15]	HRNet-W18	N	384 × 288	1.3	1.2	56.0	83.8	63.0	52.4	62.6	62.6
Lite-HRNet [11]	Lite-HRNet-18	N	384 × 288	1.1	0.4	67.6	87.8	75.0	64.5	73.7	73.7
	Lite-HRNet-30	N	384 × 288	1.8	0.7	70.4	88.7	77.7	67.5	76.3	76.2
Dite-HRNet* [12]	Dite-HRNet-18	N	384 × 288	1.1	0.4	68.3	87.7	75.4	65.0	74.7	74.0
	Dite-HRNet-30	N	384 × 288	1.8	0.7	71.5	88.9	78.2	68.2	77.7	77.2
EDite-HRNet-S	Dite-HRNet-18	N	384 × 288	1.3	0.6	68.6	87.5	75.7	65.2	74.9	74.4
EDite-HRNet-L	Dite-HRNet-18	N	384 × 288	8.8	4.14	73.6	89.4	80.5	70.3	80.2	78.9

human detection box maintains a fixed ratio of 4:3. Then the images are cropped according to the detection box and adjusted to the size of 256×192 and 384×288. In terms of MPII dataset preprocessing, the images are resized to 256×256 instead. All experiments employ data augmentation techniques including random rotation with factor 30, random scaling with factor 25 and random flippings for both COCO2017 and MPII datasets. The crowdpose dataset experiments is basically consistent with the COCO dataset experiments.

3) TESTING

In the COCO2017 dataset and crowdpose dataset, we use the human bounding box obtained by the same human detector as SimpleBaseline. Meanwhile, in the MPII dataset, a standard testing strategy is adopted for the provided person boxes. Similar to other works, we utilize 2D Gaussian heatmaps to estimate human keypoints, and use the average of original image and inverted image as the key point position. Due to the problem of quantization error, we use a deviation of 1/4 from the highest value

TABLE 2. Comparasions of results on the COCO test-dev 2017 set. The image size of the experimental test adopts two commonly used 256×192 and 384×288. The basic unit of parameter quantity is M, and the calculation quantity unit is G.

model	backbone	input size	#Params(M)	GFLOPs	AP	AP^{50}	AP^{75}	AP^{M}	AP^{L}	AR
Large networks										
Simple Baseline [2]	ResNet-50	256 × 192	34.0	8.9	70.0	90.9	77.9	66.8	75.8	75.6
CPN [27]	ResNet-Inception	384 × 288	-	-	72.1	91.4	80.0	68.7	77.2	78.5
HRNet [8]	HRNet-W32	384 × 288	28.5	16.0	74.9	92.5	82.8	71.3	80.9	80.1
UDP [28]	HRNet-W32	384 × 288	28.7	16.1	76.1	92.5	83.5	72.8	82.0	81.3
DARK [29]	HRNet-W48	384 × 288	63.6	32.9	76.2	92.5	83.6	72.5	82.4	81.8
Small networks										
Mobile NetV2 1× [19]	MobileNetV2	384 × 288	9.8	3.3	66.8	90.0	74.0	62.6	73.3	72.3
Shuffle Net V2 1× [16]	ShuffleNetV2	384 × 288	7.6	2.8	62.9	88.5	69.4	58.9	69.3	68.9
Small HRNet [15]	HRNet-W18	384 × 288	1.3	1.2	55.2	85.8	61.4	51.7	61.2	61.5
Lite-HRNet [11]	Lite-HRNet-18	384 × 288	1.1	0.4	66.9	89.4	74.4	64.0	72.2	72.6
	Lite-HRNet-30	384 × 288	1.8	0.7	69.7	90.7	77.5	66.9	75.0	75.4
Dite-HRNet* [12]	Dite-HRNet-18	384 × 288	1.1	0.4	68.4	89.9	75.8	65.2	73.8	74.4
	Dite-HRNet-30	384 × 288	1.8	0.7	70.6	90.8	78.2	67.4	76.1	76.4
EDite-HRNet-S	Dite-HRNet-18	384 × 288	1.3	0.6	69.2	90.2	76.9	66.1	74.7	74.7
EDite-HRNet-L	Dite-HRNet-18	384 × 288	8.8	4.14	72.8	91.3	80.6	69.7	78.4	78.2

TABLE 3. Comparisons of results on the MPII val set. The compared models are all lightweight pose estimation models and their deformations.

Method	#Params(M)	GFLOPs	PCKh
MobileNetV2 1× [19]	9.6	1.9	85.4
MobileNetV3 1× [19]	8.7	1.8	84.3
ShuffleNetV2 1× [16]	7.6	1.7	82.8
Small HRNet [6]	1.3	0.7	80.2
Lite-HRNet-18 [11]	1.1	0.2	86.1
Lite-HRNet-30 [11]	1.8	0.4	87.0
Dite-HRNet-18 [12]	1.1	0.2	86.6
Dite-HRNet-30 [12]	1.8	0.4	87.6
EDite-HRNet-L	9.1	2.46	88.6
EDite-HRNet-S	1.3	0.3	86.8

position to the second highest position as the final keypoint position.

B. RESULTS

1) RESULTS ON THE COCO val2017 SET

As shown in Table 1, according to the amount of parameters and computation, our method is divided into two structures EDite-HRNet-L and EDite-HRNet-S, which adopt the same network structure design. The EDite-HRNet-L structure adopts a normal convolution operation, while the EDite-HRNet-S structure uses a depth-wise convolution to further reduce the amount of parameters and calculations.

Compared with models such as HRNet, UDP, DARK, etc., the prediction accuracy of our model EDite-HRNet-S is 5% lower than them, but our calculation amount is 2.8% of theirs. Compared with lightweight models such as Lite-HRNet and Dite-HRNet, our model EDite-HRNet-S has improved the calculation accuracy by 1.3% and 0.9% at 256×192 image size. At 384×288 image size, our method is 0.3 better than Dite-HRNet.

At the image size of 256×192, our large model EDite-HRNet-L is basically the same as the SimpleBaseline prediction accuracy, but our calculation amount is only half of the original general average, and the number of parameters is only 1/4 of the original. At 384×288 image size, the large model EDite-HRNet-S surpasses all lightweight methods, and the prediction accuracy is close to HRNet. Generally, Our method achieves a better balance between model accuracy and complexity.

2) RESULTS ON THE COCO TEST-DEV SET

The COCO test focused on the method comparison generally used 384×288 image size, so we also used the same size to facilitate comparison. Compared to those large networks, our EDite-HRNet-L performance is basically similar (72.8%), but the computational load is greatly reduced in Table 2. Compared with the latest lightweight pose estimation model Dite-HRNet, our EDite-HRNet-S is 0.8% higher than the original under the same backbone network conditions (Dite-HRNet-18).

3) RESULTS ON THE MPII VAL SET

The experimental results on the MPII dataset are shown in Table 3. Our method EDite-HRNet-S is 0.2 percentage points higher than Dite-HRNet under the condition that the computational costs are similar. In terms of the lightweight design, our method surpasses other SOTA lightweight methods and achieves the best balance between accuracy and computational cost. In terms of prediction accuracy, our method EDite-HRNet-L utilizes the network structure of Dite-HRNet-18 and surpasses the model performance of Dite-HRNet-30. Since both of our methods adopt the same structure design as Dite-HRNet-18, we can flexibly design the number of group convolutions in the convolutional

TABLE 4. Comparisons of results on the crowdpose test set.

Method	Backbone	Input Size	GFLOPs	AP
Mask R-CNN [31]	ResNet	256 × 192	-	57.2
AlphaPose [32]	AlphaPose	256 × 192	-	61.0
SimpleBaseline	ResNet-50	256 × 192	5.46	63.7
HRNet	HRNet-W32	256 × 192	7.7	67.5
RTMPose [33]	CSPNetXt-m	256 × 192	1.93	66.9
Dite-HRNet	Dite-HRNet-18	256 × 192	0.4	58.5
EDite-HRNet-S	Dite-HRNet-18	256 × 192	0.6	58.5
EDite-HRNet-L	Dite-HRNet-18	256 × 192	4.14	62.9

TABLE 5. Ablation experiments on COCO val2017 and MPII dataset. The MFLOPs and Params are computed with the input size 256 × 192 ofr COCO val2017set and 256 × 256 for MPII val set,respectively.

Method	COCO2017			MPII	
	#Params(M)	MFLOPs	AP	MFLOPs	PCKh
Dite-HRNet-18*	1.1	0.2	65.2	0.2	86.6
+Enhance ConvBN	1.2	0.2	65.4	0.2	87.0
+AEDMC block	1.3	0.2	66.1	0.3	86.8

layer to achieve the best balance between performance and computational cost.

4) RESULTS ON THE CROWDPOSE TEST SET

We add a comparison of experimental results of the crowdpose dataset to the Dite-HRNet basic method. The general image size of 256×192 was used, and the trainval file was used for training and the test file was used for testing. The other settings are basically consistent with the COCO dataset.

Compared with those large networks, our EDite-HRNet-L has achieved similar prediction accuracy, but there are still gaps. Because the crowdpose treatment methods used by those methods are different to a certain extent, the experimental results are not completely fair. We compared the DIte-HRNet method and the EDite-HRNet method under the same environmental conditions, and their prediction accuracy was the same. This situation is mainly because the method in this paper is more suitable for single or small number of people, and the effect is not obvious in the case of multiple people.

C. ABLATION STUDY

To demonstrate the effectiveness of our proposed method EDite-HRNet, we conduct ablation experiments on COCO2017 and MPII datasets. We first use Dite-HRNet as the baseline. The Enhanced ConvBN block and EDMC block are then added individually or together to the baseline.

The final network model we obtained is EDite-HRNet-S. The learning rate, data preprocessing and other elements are consistent with the settings of Dite-HRNet method. In order to ensure the accuracy of the experimental results, we conduct training and testing on the same experimental platform, and the experimental results of Dite-HRNet are 0.7 lower than the experimental results in the paper.

As shown in all tables, * indicates that the same experimental platform is used for experimental comparison. Compared to the Dite-HRNet-18 network, our method increases by 0.9 percentage points on the COCO2017 dataset and increases by 0.2 percentage points on the MPII dataset. After embedding the Enhanced ConvBN block, it increased by 0.4 percentage points on the MPII dataset, and after embedding the EDMC block, it decreased by 0.2 percentage points. We speculate that the problem is caused by the different learning effects of these two blocks on the dataset. From the experimental results, our EDMC block and Enhanced ConvBN block facilitate EDite-HRNet to achieve a better balance between prediction accuracy and computational cost.

V. CONCLUSION

To improve the prediction accuracy of lightweight pose estimation, we propose the EDite-HRNet network. We redesigned the EDMC and EDGC blocks to make full use of the image features of different branch structures. Based on the redesigned blocks, we propose the EDite-HRNet network model. Under similar computational load conditions, our EDite-HRNet-S outperforms the Dite-HRNet by 0.9 and 0.2 on COCO and MPII datasets, respectively. In addition, our EDite-HRNet-L has a large increase in parameters, but the corresponding accuracy is also greatly improved by 2.1 on COCO and 2.0 on MPII. Our model can adapt to platforms with different computing capabilities according to actual needs, and achieve a better balance between prediction accuracy and computing load.

Although, our proposed method increases the prediction accuracy of the lightweight pose estimation model with a small computational load, but the training time and training complexity of the model are increased. In future work, it is necessary to design optimization strategies to accelerate the training of lightweight models.

REFERENCES

[1] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *Proc. ECCV*, vol. 9912, 2016, pp. 483–499.

[2] B. Xiao, H. Wu, and Y. Wei, "Simple baselines for human pose estimation and tracking," in *Proc. ECCV*, vol. 11210, 2018, pp. 472–487.

[3] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected CRFs," in *Proc. ICLR*, 2015, pp. 1–13.

[4] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder–decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[5] T. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie, "Feature pyramid networks for object detection," in *Proc. CVPR*, 2017, pp. 936–944.

[6] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, W. Liu, and B. Xiao, "Deep high-resolution representation learning for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 10, pp. 3349–3364, Oct. 2021.

[7] H. Wu and B. Xiao, "3D human pose estimation via explicit compositional depth maps," in *Proc. AAAI Conf. Artif. Intell.*, Apr. 2020, vol. 34, no. 7, pp. 12378–12385.

[8] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," in *Proc. CVPR*, 2019, pp. 5693–5703.

[9] H.-S. Fang, J. Li, H. Tang, C. Xu, H. Zhu, Y. Xiu, Y.-L. Li, and C. Lu, "AlphaPose: Whole-body regional multi-person pose estimation and tracking in real-time," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 6, pp. 7157–7173, Jun. 2023.

[10] Z. Kan, S. Chen, C. Zhang, Y. Tang, and Z. He, "Self-correctable and adaptable inference for generalizable human pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 5537–5546.

[11] C. Yu, B. Xiao, C. Gao, L. Yuan, L. Zhang, N. Sang, and J. Wang, "Lite-HRNet: A lightweight high-resolution network," in *Proc. CVPR*, 2021, pp. 10440–10450.

[12] Q. Li, Z. Zhang, F. Xiao, F. Zhang, and B. Bhanu, "Dite-HRNet: Dynamic lightweight high-resolution network for human pose estimation," in *Proc. IJCAI*, 2022, pp. 1095–1101.

[13] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.

[14] A. Howard, M. Sandler, B. Chen, W. Wang, L.-C. Chen, M. Tan, G. Chu, V. Vasudevan, Y. Zhu, R. Pang, H. Adam, and Q. Le, "Searching for MobileNetV3," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1314–1324.

[15] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *Proc. CVPR*, 2018, pp. 6848–6856.

[16] N. Ma, X. Zhang, H. Zheng, and J. Sun, "ShuffleNet V2: Practical guidelines for efficient CNN architecture design," in *Proc. ECCV*, vol. 11218, 2018, pp. 122–138.

[17] Y. Tang, K. Han, J. Guo, C. Xu, C. Xu, and Y. Wang, "GhostNetV2: Enhance cheap operation with long-range attention," 2022, *arXiv:2211.12905*.

[18] J. Hu, L. Shen, and G. Sun, "Squeeze-and-Excitation networks," in *Proc. CVPR*, 2018, pp. 7132–7141.

[19] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520.

[20] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu, "GhostNet: More features from cheap operations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1577–1586.

[21] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7794–7803.

[22] Y. Cao, J. Xu, S. Lin, F. Wei, and H. Hu, "GCNet: Non-local networks meet squeeze-excitation networks and beyond," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 1971–1980.

[23] C. Cui, T. Gao, S. Wei, Y. Du, R. Guo, S. Dong, B. Lu, Y. Zhou, X. Lv, Q. Liu, X. Hu, D. Yu, and Y. Ma, "PP-LCNet: A lightweight CPU convolutional neural network," 2021, *arXiv:2109.15099*.

[24] Y. Chen, X. Dai, M. Liu, D. Chen, L. Yuan, and Z. Liu, "Dynamic convolution: Attention over convolution kernels," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11027–11036.

[25] B. Yang, G. Bender, Q. V. Le, and J. Ngiam, "CondConv: Conditionally parameterized convolutions for efficient inference," in *Proc. NeurIPS*, 2019, pp. 1305–1316.

[26] K. Han, Y. Wang, C. Xu, J. Guo, C. Xu, E. Wu, and Q. Tian, "GhostNets on heterogeneous devices via cheap operations," *Int. J. Comput. Vis.*, vol. 130, no. 4, pp. 1050–1069, Apr. 2022.

[27] Y. Chen, Z. Wang, Y. Peng, Z. Zhang, G. Yu, and J. Sun, "Cascaded pyramid network for multi-person pose estimation," in *Proc. CVPR*, 2018, pp. 7103–7112.

[28] J. Huang, Z. Zhu, F. Guo, and G. Huang, "The devil is in the details: Delving into unbiased data processing for human pose estimation," in *Proc. CVPR*, 2020, pp. 5699–5708.

[29] F. Zhang, X. Zhu, H. Dai, M. Ye, and C. Zhu, "Distribution-aware coordinate representation for human pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 7091–7100.

[30] Y. Chen, X. Dai, M. Liu, D. Chen, L. Yuan, and Z. Liu, "Dynamic ReLU," in *Proc. ECCV*, vol. 12364, 2020, pp. 351–367.

[31] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 386–397, Feb. 2020.

[32] H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu, "RMPE: Regional multi-person pose estimation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2017, pp. 2353–2362.

[33] T. Jiang, P. Lu, L. Zhang, N. Ma, R. Han, C. Lyu, Y. Li, and K. Chen, "RTMPose: Real-time multi-person pose estimation based on MmPose," 2023, *arXiv:2303.07399*.

LIYUHENG RUI was born in Sanmenxia, Henan, China, in 2001. She received the degree in computer science and technology from Northeast Forestry University, in 2019. She is currently pursuing the degree with Beihang University.

YANYAN GAO received the B.S. degree from the School of Computer Science and Technology, OEC, Hebei, China, in 2009. Since 2018, he has been working with the Human-Computer Interaction Laboratory, School of Computer Science and Technology, Beijing Institute of Technology, Beijing, China. His current research interests include embedded systems, computer vision, and object detection.

HAOPAN REN received the B.S. degree in mining engineering from Henan Polytechnic University, Henan, China, in 2018. Since 2018, he has been with the Human-Computer Interaction Laboratory, School of Computer Science and Technology, Beijing Institute of Technology, Beijing, China. His current research interests include machine learning, computer vision, object detection, and human and hand pose estimation.