Applied Intelligence (2023) 53:2161–2172
https://doi.org/10.1007/s10489-022-03623-z

Decision-level information fusion powered human pose estimation

Yiqing Zhang[1] · **Weiting Chen**[1]

Accepted: 10 April 2022 / Published online: 5 May 2022

Abstract
Human pose estimation is viewed as a crucial step for understanding human behaviour. Although significant progress has been made in this area in recent years, most studies have focused on feature-level information fusion, while decision-level information fusion has rarely been explored. Compared with feature-level information, decision-level information contains more semantic and interpretable information and can help improve the performance of pose estimation in occluded and crowded scenes. In this paper, we focus on the fusion of decision-level information. We propose a View Fusion module for aggregating decision-level information from different stages to generate a more comprehensive estimation. An Auxiliary Task module is introduced to bridge the gap between the feature extractor and the View Fusion module and to provide prior information about the form of the decision-level information. Considering that the precision of predictions from different stages varies, we use different strategies to guide the learning process. Experiments show that our models outperform previous methods and achieve competitive results on the CrowdPose test set. Further experiments indicate that our method is flexible and can improve the performance of various backbones.

Keywords Pose estimation · Information fusion · Decision-level information

1 Introduction

Human pose estimation is the determination of the posture of the human body from input images or video sequences by estimation the spatial coordinates of human joints [1]. It is widely viewed as a crucial step in understanding human behaviour, as it provides geometric and motion information for many downstream applications, including action recognition [2–4], person re-identification [5, 6], human-computer interaction [7], artificial intelligence [8, 9], robot vision [10], and intelligent control [11], among others. With the successful application of deep learning in image classification [12] and natural language processing [13, 14], deep learning based human pose estimation methods have been the dominant approach in this field and significant progress has been made.

✉ Weiting Chen
wtchen@sei.ecnu.edu.cn

Yiqing Zhang
51194501084@stu.ecnu.edu.cn

[1] MOE Research Center of Software/Hardware Co-Design Engineering, East China Normal University, Shanghai, China

Deep learning based human pose estimation methods can be categorized according to methodology into two groups: bottom-up methods and top-down methods. Bottom-up methods [15–18] first detect all keypoints in the input image and then group those keypoints into human instances, while top-down methods [19–25] first detect all human instances in the input image and then estimate the keypoints of each human instance. Bottom-up methods have advantages in terms of inference speed and have the potential to achieve real-time pose estimation since no human detector is required. However, the variation of the human scale is a major unsolved challenge in bottom-up methods, which hinders further performance improvement. In contrast, with the help of a human detector, top-down methods can reduce the multi-person pose estimation problem to a simpler single-person pose estimation problem and easily normalize all human instances to the same scale. Recent studies show that higher performance is achieved by top-down methods. In this paper, we focus on top-down methods.

Extensive studies have been conducted to design feature extraction and feature fusion methods to obtain semantic and representative information, which is essential for accurate human pose estimation. Stacked Hourglass was proposed in [21] for extracting features in a repeated encoding-decoding strategy. In addition to using the features

from the end of the previous stage as in [21], features at different resolutions were integrated thoroughly to refine the estimation in Cascaded Pyramid Network [26]. High-Resolution Net [25] (abbreviated as HRNet) was proposed for maintaining the high-resolution features through the whole process, where features from different resolutions were fused repeatedly. Apart from the well-studied inter-level feature fusion, intra-level feature fusion was proposed in RSN [27] to further improve the quality of the local representation.

Although feature-level information fusion has been well studied in human pose estimation, decision-level information fusion, which is another type of information fusion, has rarely been explored in this area. Compared with feature-level information, decision-level information contains much high-level semantic information, which is crucial for improving the performance on various tasks. Usually, decision-level information is obtained from different sources, such as different models [28], different levels [29], different sensors [30], and so on. Fusion of this information can leverage the advantages of decisions from different sources. Moreover, decision-level information is more straightforward and more interpretable than feature-level information and is easier to analyze than high-dimension semantic features. As decision-level information contains much semantic and contextual information, which is important for keypoint estimation in occluded and crowded scenes, fusing decision-level information is a promising approach and merits exploration in human pose estimation. Since human pose estimation is based on keypoint locations, the information that contains the locations of the keypoints can be viewed as decision-level information in human pose estimation. For example, the coordinates of keypoints and the keypoint heatmap can be viewed as decision-level information.

To improve the performance of human pose estimation, we propose a View Fusion module for fusing decision-level information. An Auxiliary Task module is introduced to improve the quality of the decision-level information. Our experimental results indicate the effectiveness, efficiency, and generalization of our approach.

To the best of our knowledge, we are the first to explore decision-level information fusion in human pose estimation. We highlight our contributions as follows:

1. To fully utilize the decision-level information, we propose a View Fusion module for generating a comprehensive estimation by integrating decision-level information from each stage.
2. An Auxiliary task module is introduced to bridge the gap between the feature extractor and the View Fusion module and to provide prior information about the form of the decision-level information.
3. Our method can be easily incorporated into various backbones that are used in human pose estimation and improve the performance with negligible additional parameters and computational cost.

The remainder of this paper is organized as follows: Section 2 reviews the related works. Section 3 describes our proposed method in detail. Section 4 presents and analyzes our experimental results and ablation studies. Section 5 concludes the paper.

2 Related works

2.1 Heatmap regression

DeepPose [19], which was the first method in which deep learning was introduced into human pose estimation, regards the human pose estimation task as a coordinate regression problem, where the output of the model is the coordinates of the keypoints. Since the positions near the exact locations of the keypoints can also be viewed as correct predictions, predicting the exact locations is not necessary. To this end, keypoints heatmaps were used as the output of the model in [20] to relax the constraint, which greatly improved the performance. Since then, heatmap regression has become the mainstream methodology for human pose estimation. Despite the good performance, the heatmap regression is not differentiable because it contains *argmax*, which is used to select the location with the largest heat value. To overcome this, an integral operation was proposed in [31] to relate and unify heatmap regression and coordinate regression. Following this work, several approaches for combining heatmap regression and coordinate regression have been proposed [24, 32].

Since heatmap regression has become the most prevalent approach for human pose estimation and shows better performance, in this paper, we focus on heatmap regression methods.

2.2 Top-down human pose estimation pipeline

The pipeline of top-down human pose estimation is introduced for reference. To estimate human pose, a human detector is applied to obtain the bounding boxes of target humans. These bounding boxes are used to crop the target humans from the original image. For each cropped image, the output of the model is N heatmaps of size $H' \times W'$, where N is the number of keypoints. The heat value of the nth heatmap indicates the location confidence of the nth keypoint (e.g., if the nth keypoint is the left shoulder, then the heat value of this heatmap indicates the location confidence of the left shoulder). Post-processing is applied to the

heatmaps to obtain the spatial coordinates of each keypoint. A common strategy is to choose the location of the largest heat value as the location of the keypoint.

In the training stage, the bounding boxes are obtained from the annotation files. The loss function is defined on the predicted heatmaps and the ground truth heatmaps. The ground truth heatmaps are generated by applying Gaussian blur to the annotated joint coordinates. Formally, the value of (x, y) in the heatmap, denoted as $G(x, y)$, is defined in (1),

$$G(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(x-u)^2 + (y-v)^2}{2\sigma^2}\right) \tag{1}$$

where $\exp(\cdot)$ is an exponential function, σ is the variance and (u, v) is the location of the keypoint.

The choice of σ affects the accuracy of human pose estimation. Ground truth heatmaps that are generated using various values of σ are shown in Fig. 1. The pixel in red corresponds to the highest value and the pixel in blue to the lowest value. The smaller σ is, the fewer pixels with high value there are in the heatmap. The value of α can affect the performance of keypoint detection. (see Appendix A)

2.3 Information fusion

Since representative features are important for obtaining accurate pose estimation, extensive studies have been conducted to explore how to fuse features from different scales and different stages. RefineNet was proposed in [26] for transmitting information across different levels. An intra- and inter-feature extraction module was proposed in [27] for retaining rich low-level spatial information. Neural architecture search was used in [33] to find an efficient backbone for human pose estimation. Repeated parallel multi-resolution fusion was used in [25] to enrich the high-resolu tion features.

In human pose estimation, although feature-level information fusion has been well studied, decision-level information fusion has rarely been explored. Decision-level information fusion has been successfully used in many fields [29, 34, 35]. Since much semantic and contextual information is encoded in the decision-level information, in this paper, we explore how to fuse this information to improve the accuracy of human pose estimation. To the best of our knowledge, we are the first to explore decision-level information fusion in human pose estimation.

3 Approach

As illustrated in Fig. 2, our method consists of three parts: an Encoder, an Auxiliary Task module, and a View Fusion module. The Encoder extracts features from the input image. The Auxiliary Task module is added to bridge the gap between the Encoder and the View Fusion module and to provide prior information. Different strategies are used to guide the model to gradually focus on improving the estimation of hard keypoints. The View Fusion module is designed to fuse decision-level information. This module is responsible for producing a comprehensive estimation by integrating and balancing decision-level information generated from different stages. We will introduce each part in detail in the following paragraphs.

3.1 Encoder

The Encoder is used to extract features from the input image. Note that our method is independent of the backbone used as the Encoder and can be easily incorporated in various backbones.

We use HRNet [25], of which the architecture is surrounded by a red box in Fig. 2, as an example to demonstrate

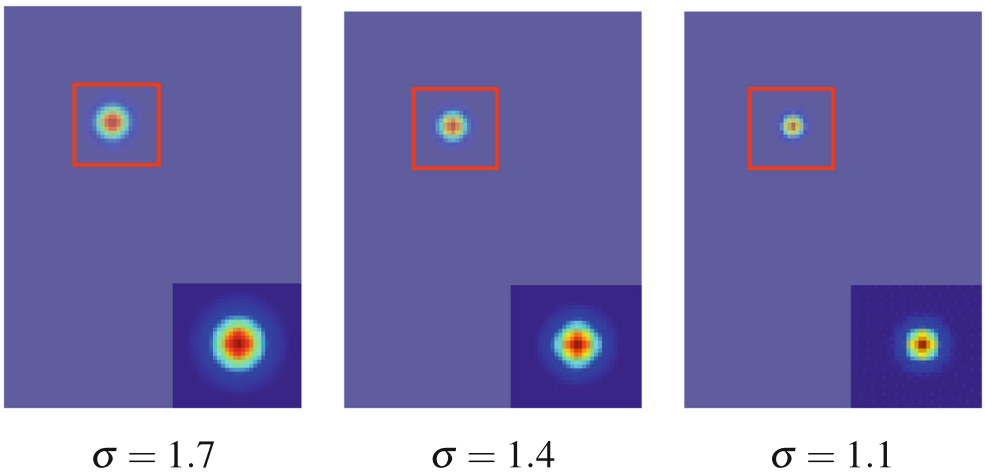


Fig. 1 Ground-truth heatmaps that are generated with various values of σ

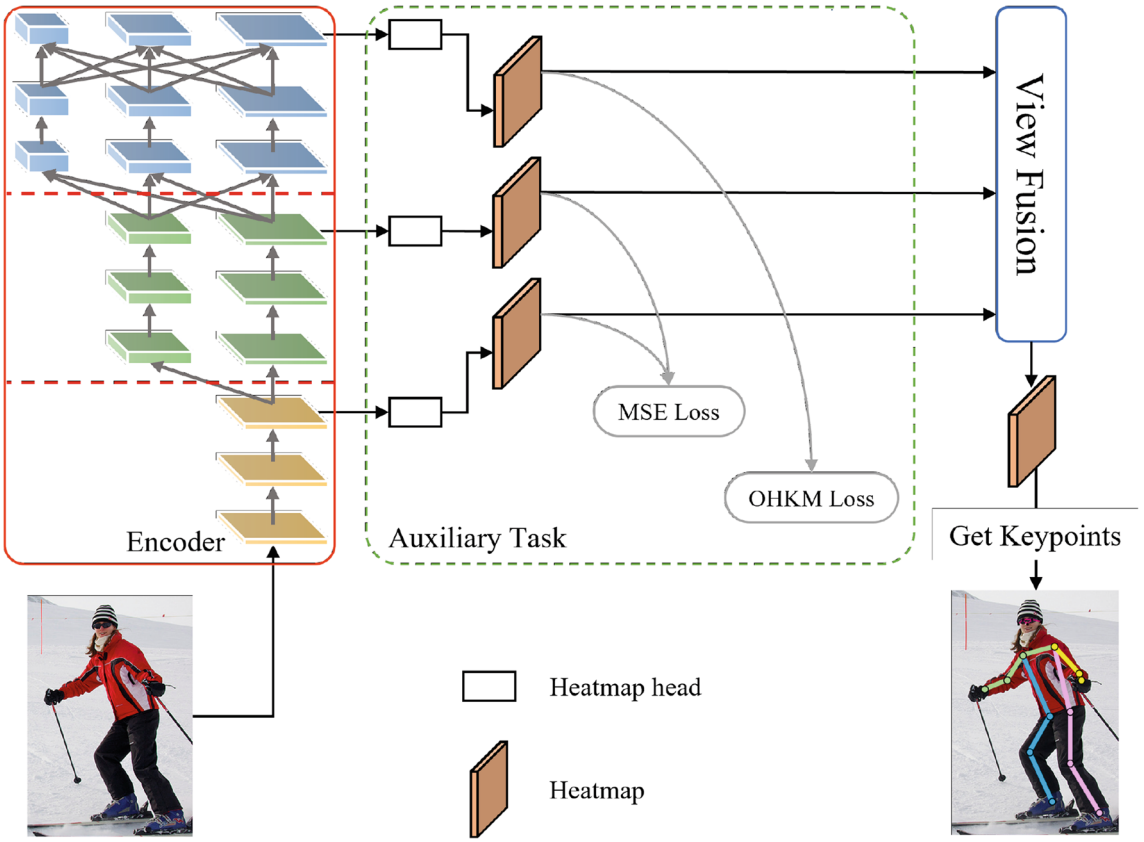


Fig. 2 Overview of our proposed method. The MSE denotes Mean Square Error and the OHKM stands for Online Hard Keypoint Mining

the application of our method. Based on the finding that high-resolution features are crucial for obtaining an accurate pose estimation, HRNet was designed to maintain high-resolution features throughout the whole training process. A repeated multi-resolution fusion schema was used in HRNet to enrich the high-resolution features.

Since a deep neural network is a stack of basic operations, such as convolution, pooling, linear, etc., we can easily divide it into several stages. As illustrated in Fig. 2, we can divide HRNet into several stages, and each stage is distinguished by a different colour.

Formally, the Encoder can be split into K stages, each of which is denoted as Enc_k. Given an input image I of size $C \times H \times W$, the feature that is extracted by each stage can be expressed as (2).

$$F_k = Enc_k(I) \tag{2}$$

3.2 View fusion module

To integrate decision-level information from different stages, we propose the View Fusion module.

Instead of the feature-level aggregation that was studied in previous works, we focus on integrating decision-level information. Since each value in the keypoint heatmap represents the probabilities of the corresponding pixel being a keypoint, in this work, we use that as the decision-level information.

Decision-level information is high-level information that contains rich semantic and contextual information, which can help to improve the accuracy of keypoint estimation in occluded and crowded scenes. Decision-level information can help distinguish the type of occluded keypoints, determine the belonging of the keypoints, and produce an estimation that matches the natural structure of the human body.

From the perspective of the source of decision-level information, we obtain decision-level information from different stages of the Encoder. Information from different stages introduces different views of the input data. 1) Different stages of the network focus on different scales of information, from local information to global information. 2) Different levels of information are extracted by different stages of the models. Low-level local information is extracted in the earlier stages, and high-level semantic and contextual information is extracted in the latter stages. Aggregation of all these kinds of information can leverage the advantages of each view and help the model to realize more comprehensive pose estimation.

The architecture of the View Fusion module is illustrated in Fig. 3. The decision-level information generated by the Auxiliary Task module is fed into the View Fusion module. We will present the Auxiliary Task module in detail in Section 3.3. Formally, the output of the Auxiliary Task module is denoted as $A = [A_1, A_2, \cdots, A_K]$, where A_i is decision-level information of size $N \times H \times W$ and K is the number of stages in the Encoder. We stack H_i along axis 1 and reshape it into a tensor of size $K \times N \times Z$, where $Z = H \times W$. Then, this tensor is fed into a Conv-BatchNorm-ReLU module, which is a sequential stack of a convolution layer, a BatchNorm layer, and a ReLU activiateion layer. In the Conv-BatchNorm-ReLU module, the convolution, Conv2d(in_channels=K, out_channels=M, kernel_size=1, stride=1), is designed to extract information from all collected decision-level information by using M different kernels. Then, the output of this convolution is sent to a BatchNorm layer and activated by ReLU. Finally, a convolution, Conv2d(in_channels=M, out_channels=1, kernel_size=1, stride=1), is employed to obtain the final prediction by projecting to $1 \times N \times Z$. The keypoint heatmap of size $N \times H \times W$ is obtained after a reshaping operation.

3.3 Auxiliary task module

To bridge the gap between the Encoder and the View Fusion module and to provide prior information, we introduce the Auxiliary Task module, which can also disentangle the optimization process.

The View Fusion module is designed to integrate and balance decision-level information, while the Encoder aims at extracting features. There is a gap between the View Fusion module and the Encoder. To bridge this gap, we use heatmap heads in the Auxiliary Task module to transform the features extracted by the Encoder into decision-level information. Then, we present the design of the heatmap head in detail. Given a feature F_i of size $C \times H' \times W'$ extracted by stage i, an upsample module is applied to match the resolution of the feature with the resolution of the output. Then, a convolution, namely, Conv2d(in_channels=C, out_channels=N, kernel_size=1, stride=1), activated by ReLU is used to get the output decision-level information A_i of size $N \times H \times W$, where N is the number of keypoints. The whole process can be written as (3).

$$A_i = ReLU(Conv(Upsample(F_i))) \tag{3}$$

On the other hand, a good form of decision-level information is helpful to improve the accuracy of human pose estimation. However, the form of decision-level information found by the model itself may be not suitable enough and even hinder the performance. To solve this problem, we use the loss functions introduced in the Auxiliary Task module to provide prior information about the form of the decision-level information and encourage the heatmap heads to transform the features to the desired form of decision-level information. With this prior information, we expect the model to focus on extracting better decision-level information. Besides, the Auxiliary Task module can disentangle the optimization process and help the model training as the intermediate supervision.

We illustrate the process of the Auxiliary Task module in the dotted green box in Fig. 2. The loss functions in the Auxiliary Task module are defined between the ground truth keypoint heatmaps and the outputs of the heatmap heads. Two loss functions are used to guide the latter stage of the model to giving more attention to keypoints that are poorly predicted. The detail of the loss functions are presented in Section 3.4.

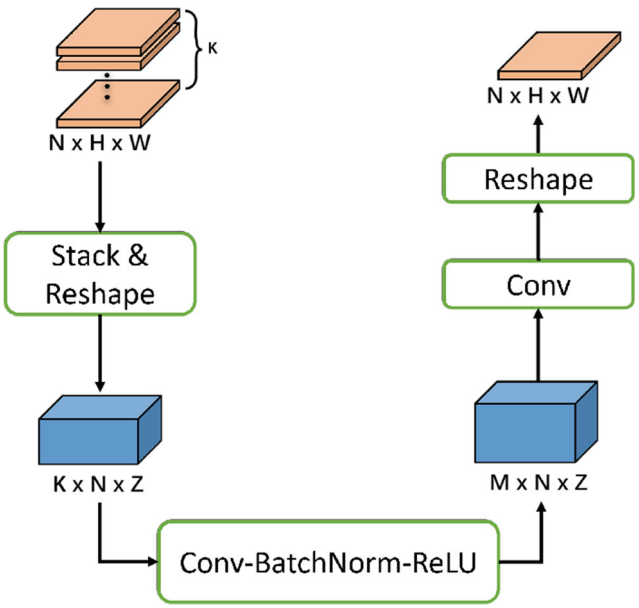


Fig. 3 Vie w Fusion module

3.4 Loss function

The overall loss function of our model is a weighted sum of the loss over all predictions, namely, the predictions from each stage and one prediction from the View Fusion module. Considering that the precision of predictions from different stages varies, we use different loss functions to guide the learning process.

For a prediction from stage i, we define the loss function L_i as in (4),

$$L_i = D_i(H_i, GT) \tag{4}$$

where GT is the ground truth keypoint heatmap and H_i is the output heatmap of stage i. Note that all the L_i, where i belongs to $[1, \cdots, K]$, are the corresponding loss functions introduced by the Auxiliary Task module.

For the prediction from the View Fusion module, we define the loss as in (5),

$$L_{VF} = D_{VF}\left(\phi\left(H_1, \cdots, H_K\right), GT\right) \tag{5}$$

where ϕ denotes the View Fusion module.

D_i and D_{VF} can be any function that measure the quality of the predictions and can be changed according to the task. In this paper, we use two different functions. The first one is the Mean Square Error. The loss function L can be written as L_{MSE} in (6),

$$L_{MSE} = \frac{1}{N}\sum_{j=1}^{N}\left\|H^j - GT^j\right\|^2 \tag{6}$$

where H^j is the estimated heatmap of jth keypoint and GT^j is the ground truth heatmap of jth keypoint. The other one is the Online Hard Keypoint Mining [26] (abbreviated as OHKM) , which optimizes the model using only the top m keypoints losses for each human instance. In this case, the loss function L can be written as L_{OHKM} in (7).

$$\begin{aligned} L^j &= \|H^j - GT^j\|^2 \\ L_{OHKM} &= \frac{1}{m}\sum_{j\in\{j|L^j\in topk(L^j), j\in[1,N]\}} L^j \end{aligned} \tag{7}$$

The MSE treats all keypoints equally, while the OHKM forces the model to giving more attention to the hard keypoints, of which the loss values are relatively higher than others. Considering that the predictions from the latter stages of the model are usually more precise than from the earlier stages, we think that giving more attention to hard keypoints could help the training of the latter stages of the model. Thus, the OHKM is used for the final stage of the Encoder and the View Fusion module, while the MSE is used for all stages of the Encoder except the last stage.

Finally, the overall loss function of our model can be written as (8),

$$L_{model} = \sum_{i=1}^{K}\alpha_i L_i + \beta L_{VF} \tag{8}$$

where α_k and β are the weights of the terms.

Since predictions from earlier stages of the model are usually coarse predictions and may contain noise (or even error), giving equal attention to all predictions is not appropriate. To make the model give more attention to the precise prediction, we use a larger weight for the latter stage of the model. Thus, we balance the benefit of integrating predictions from all stages and the impact of noisy information.

4 Experiment

4.1 Dataset

We evaluated our approach on two public datasets: COCO [36] and CrowdPose [37].

4.1.1 COCO dataset

COCO is a widely used, large-scale, and challenging dataset that contains over 200k images and 250k person instances, which are each labelled with 17 keypoints. We trained our model on COCO train2017, which includes 57k images and 150k person instances. We evaluated the model performance on COCO val2017 and test-dev2017.

4.1.2 CrowdPose dataset

CrowdPose is a new challenging benchmark that was designed to improve the performance in crowded scenes. It contains more crowded scenes and annotated invisible keypoints than COCO. It provides 20k images and 80k person instances for training and evaluation. We followed the settings that were used in [17, 38], in which train and validation sets were used for training and the test set was used for evaluation.

4.2 Evaluation metrics

We used the average precision (AP) and average recall (AR) with various object scales and Object Keypoint Similarity(OKS) thresholds, which have been widely used to measure the performance of human pose estimation, to evaluate our method. OKS, which is defined in (9), can be regarded as an IoU-like metric for pose estimation.

$$OKS_p = \frac{\sum_i^N \exp(\frac{-d_{pi}^2}{2S_p^2k_i^2})\delta(v_{pi}=1)}{\sum_i^N \delta(v_{pi}=1)} \tag{9}$$

For each person instance p, d_{pi} is the Euclidean distance of the estimated keypoint i to its corresponding ground truth. N is the number of keypoints. $\exp(\cdot)$ is an exponential function. $\delta(\cdot)$ is an indicator function. v_{pi} is the visibility of keypoint i. S_p is the object scale. k_i is a per-keypoint constant that controls the falloff. The keypoint similarities are averaged over all labelled keypoints.

To better evaluate the performance in various crowded scenes, we also used the average precision with various values of the *crowd index*, namely, easy (0-0.1), medium (0.1-0.8), and hard (0.8-1), as defined in the CrowdPose dataset.

4.3 Implementation details

We followed the training procedure that was used in [25]. We extended the human detection box in height or width to a fixed aspect ratio of height:width = 4:3 and cropped the box from the image, which was resized to the same size, namely, 256 × 192 or 384 x 288 in our settings. The data augmentation included random rotation ($[-45°, +45°]$), random scaling ([0.65, 1.35]), and random flipping. Following [25], half-body data augmentation was also applied. Our models were implemented in PyTorch [46]. We used HRNet-w32 as our encoder if not otherwise specified.

We used the Adam [47] optimizer to train the model. The base learning rate was 5e-4. It was decreased to 5e-5 at 170 epochs and to 5e-6 at 200 epochs. The training process was terminated within 210 epochs. All experiments were conducted on one TITAN XP with a batch size of 32. For the weights in the loss function, we empirically set $\alpha_1 = \alpha_2 = 0.25, \alpha_3 = 1, \beta = 1$ in all our experiments. We will discuss the choice of loss weights in Section 4.6.1.

In a top-down method, human instances in each image should be detected first. The performance of the human detector has an impact on the pose estimation model. Thus, for fair comparison, it is important to make the human detection results consistent across the compared models. For the CrowdPose dataset, we used the human detection result that was provided by [37]. For the COCO dataset, we used the human detection result that was provided by [25]. A flip test, in which we averaged the heatmaps of the original and flipped images, was used. We adopted the same strategy as in [25] to obtain the coordinates of each keypoint.

4.4 Comparison with the state-of-the-art methods

We present the comparison results in Table 1. For fair comparison, we used the same human instance detection results across all top-down models.

Our model, using HRNet w32 as the backbone and 256x192 as the input size, achieved 68.8 AP on the CrowdPose test set, which surpassed all the compared methods using the same input size.

Our model can achieve better performance when using a larger input size. After enlarging the input size to 384x288, AP was further boosted from 68.8 to 70.0. This model achieved the second best AP on the CrowdPose test set. Although OPECNet achieved a slightly higher AP than our model, we suppose the main reasons are: 1) A larger backbone, ResNet 101, is used in OPECNet. 2) OPECNet is a two stage method which designs a GCN module to refine pose.

Comparing the last three columns of Table 1, we find that our method outperformed all the previous models in terms of AP on all kinds of crowded scenes. The gains in AP(E) and AP(M) were 1.6 and 2.3, respectively. Although a larger backbone and larger input size were used in DEKR, which achieved the second best AP(H) of 58.7, our model still outperformed it with a gain of 0.7.

We also present the experimental results on COCO val2017 and test-dev2017 in Table 2. We observe consistent improvement in AP on both COCO val2017 and test-dev2017. Since uncrowded scenes dominate in the COCO dataset and many invisible keypoints are not annotated, the improvement in the AP score is relatively small.

The experimental results that are presented above imply that our method can improve the performance of human pose estimation in both normal scenes and crowded scenes.

4.5 Generalization of our method

To verify the generalization of our method, we changed the Encoder from HRNet to SimpleBaseline [22], a simple and famous method for human pose estimation, and to LiteHRNet [48], a leading lightweight model in human pose estimation. Since the backbone of SimpleBaseline is ResNet 50 [49], in which feature resolutions differ among stages, we used transpose convolution to upsample the features to match the resolution of the decision-level information before sending them to the Auxiliary Task module. As the feature resolutions from different stages are consistent in LiteHRNet, no further processing is needed. Evaluation results on the CrowdPose test set are presented in Table 3. We find that, in addition to HRNet, our method also performed effectively on SimpleBaseline and LiteHRNet, improving the AP score by 1.5 and 2.2, respectively. These results indicate that our approach is flexible and can improve the performance of various backbones.

4.6 Empirical study

We conducted empirical studies on the CrowdPose dataset to investigate the effectiveness of our approach and provide insight into why our approach works.

4.6.1 Choice of loss weight

We present the evaluation results that were obtaind using various loss weights in Table 4. We used LiteHRNet-w18 as the Encoder. The input size was 256x192. In experiment A, we set all weights to 1. In experiment B and experiment C, we reduced α_1 and α_2, which are the loss weights of the earlier stages, to 0.5 and 0.25, respectively. We find that the AP score improved after α_1 and α_2 were decreased. This observation confirms that giving equal attention to each prediction hinders the performance. Thus, we used a larger weight for the later stage and a smaller weight for the earlier stage to balance the benefit of integrating prediction from all stages and the impact of noisy information.

Table 1 Comparison results on the CrowdPose test set

Model	Year	Input Size	Backbone	AP	$AP^{.50}$	$AP^{.75}$	AR	$AR^{.50}$	$AR^{.75}$	AP(E)	AP(M)	AP(H)
Bottom-up Methods												
OpenPose [39]	TPAMI-2019			-	-	-	-	-	-	62.7	48.7	32.3
HigherHRNet [17]	CVPR-2020	640x640	HRNet w48	65.9	86.4	70.6	-	-	-	73.3	66.5	57.9
DEKR [38]	CVPR-2021	640x640	HRNet w48	67.3	**86.4**	72.2	-	-	-	74.6	68.1	58.7
Top-down Methods												
HOPE [40]	JAIHC-2021	384×288	ResNet 101	56.3	77.4	61.0	60.8	78.7	78.7	-	-	-
Mask-RCNN [41]	TPAMI-2020			57.2	83.5	60.3	65.9	89.5	69.4	69.4	57.9	45.8
AlphaPose [42]	ICCV-2017			61.0	81.3	66.0	67.6	86.7	71.8	71.2	61.4	51.1
SimpleBaseline* [22]	ECCV-2018	256x192	ResNet 50	62.9	80.5	68.6	72.6	90.7	78.1	73.7	64.1	50.0
CFENet [43]	APIN-2021	256x192	ResNet 152	64.2	82.2	69.5	70.7	87.7	75.5	73.6	63.7	55.6
SPPE [37]	CVPR-2019	320x256	ResNet 101	66.0	84.2	71.5	72.7	89.5	77.5	75.5	66.3	57.4
HRNet* [25]	TPAMI-2021	256x192	HRNet w32	67.5	82.5	72.9	76.3	91.6	81.6	76.9	68.7	55.4
MIPNet [44]	AAAI-2021	384x288	HRNet w48	70.0	-	-	-	-	-	-	-	-
OPECNet [45]	ECCV-2020	320x256	ResNet 101	**70.6**	**86.8**	**75.6**	-	-	-	-	-	-
Ours		256x192	HRNet w32	68.8	83.3	74.4	77.1	**91.7**	82.1	77.5	69.9	57.5
Ours		384x288	HRNet w32	70.0	84.3	75.3	**77.5**	91.6	**82.3**	**78.5**	**71.0**	**59.4**

* indicates that the results were obtained by us

Table 2 Results on COCO val2017 and test-dev2017

Model	Backbone	Input Size	val2017			test-dev2017		
			AP	AP (M)	AP (L)	AP	AP (M)	AP (L)
SimpleBaseline [22]	ResNet 50	256x192	71.8	67.9	78.6	70.0	71.5	81.3
Ours	ResNet 50	256x192	**72.4**	**69.0**	**78.8**	**71.6**	**72.9**	**82.7**
HRNet [25]	HRNet w32	256x192	74.4	70.8	81.0	73.5	75.0	84.5
Ours	HRNet w32	256x192	**75.0**	**71.3**	**82.0**	**73.8**	**75.2**	**84.7**

Table 3 Generalization of the proposed method

Model	Backbone	Input Size	AP	$AP^{.50}$	$AP^{.75}$	AR	$AR^{.50}$	$AR^{.75}$	AP(E)	AP(M)	AP(H)
HRNet [25]	HRNet w32	256x192	67.5	82.5	72.9	76.3	91.6	81.6	76.9	68.7	55.4
Ours	HRNet w32	256x192	**68.8**	**83.3**	**74.4**	**77.1**	**91.7**	**82.1**	**77.5**	**69.9**	**57.5**
LiteHRNet [48]	LiteHRNet w18	256x192	53.7	76.0	58.5	64.3	87.9	69.0	64.5	54.9	40.6
Ours	LiteHRNet w18	256x192	**55.9**	**77.6**	**60.4**	**65.4**	**88.0**	**70.0**	**66.0**	**57.2**	**43.3**
SimpleBaseline [22]	ResNet 50	256x192	62.9	80.5	68.6	72.6	90.7	78.1	73.7	64.1	50.0
Ours	ResNet 50	256x192	**64.6**	**81.7**	**70.0**	**73.4**	**90.9**	**78.6**	**74.3**	**65.7**	**52.3**

Table 4 Results of various loss weights on the CrowdPose test set

ID	α_1	α_2	α_3	β	AP	$AP^{.50}$	$AP^{.75}$	AP(E)	AP(M)	AP(H)
A	1	1	1	1	55.1	77.6	59.5	65.4	56.2	42.3
B	0.5	0.5	1	1	55.3	77.6	59.7	65.8	56.7	42.6
C	0.25	0.25	1	1	**55.9**	**77.6**	**60.4**	**66.0**	**57.2**	**43.3**

Table 5 Results of the ablation study on the CrowdPose test set

Components			AP	$AP^{.50}$	$AP^{.75}$	AP(E)	AP(M)	AP(H)	#Params	GFLOPs
HRNet w32	View Fusion	Auxiliary Task								
✓			67.5	82.5	72.9	76.9	68.7	55.4	28.536M	7.704
✓	✓		67.9	82.8	73.3	76.8	69.1	56.3	28.551M	7.751
✓	✓	✓	**68.8**	**83.3**	**74.4**	**77.5**	**69.9**	**57.5**	28.554M	7.759

4.6.2 Ablation study

We studied the effects of the View Fusion module and the Auxiliary Task module. The results are presented in Table 5. 1) Compared with the baseline model, the AP score was improved by 0.4 after the addition of the View Fusion module. 2) The Auxiliary Task module further improved the AP score by 0.9, as presented in the last row. The total improvement in the AP score exceeded 1.3, which is a considerable margin.

In addition, we investigated the source of the performance gain. Comparing the baseline with our full method, we find that our method realized AP score improvements of 0.6, 1.2 and 2.1 in uncrowded, moderately crowded and extremely crowded scenes, respectively. We conclude that the performance gain originated mostly from crowded scenes.

To evaluate the efficiency of our method, we list the total number of model parameters and the computational cost (in FLOPs) in the last two columns in Table 5. The View Fusion module brought 0.015M additional parameters and the Auxiliary Task module brought less than 0.003M additional parameters. The overall additional computational cost was approximately 0.055 GFLOPs.

In conclusion, the ablation results indicate that our proposed method boosts the model performance considerably, especially in crowded scenes, with nearly negligible increments in the numbers of parameters and FLOPs.

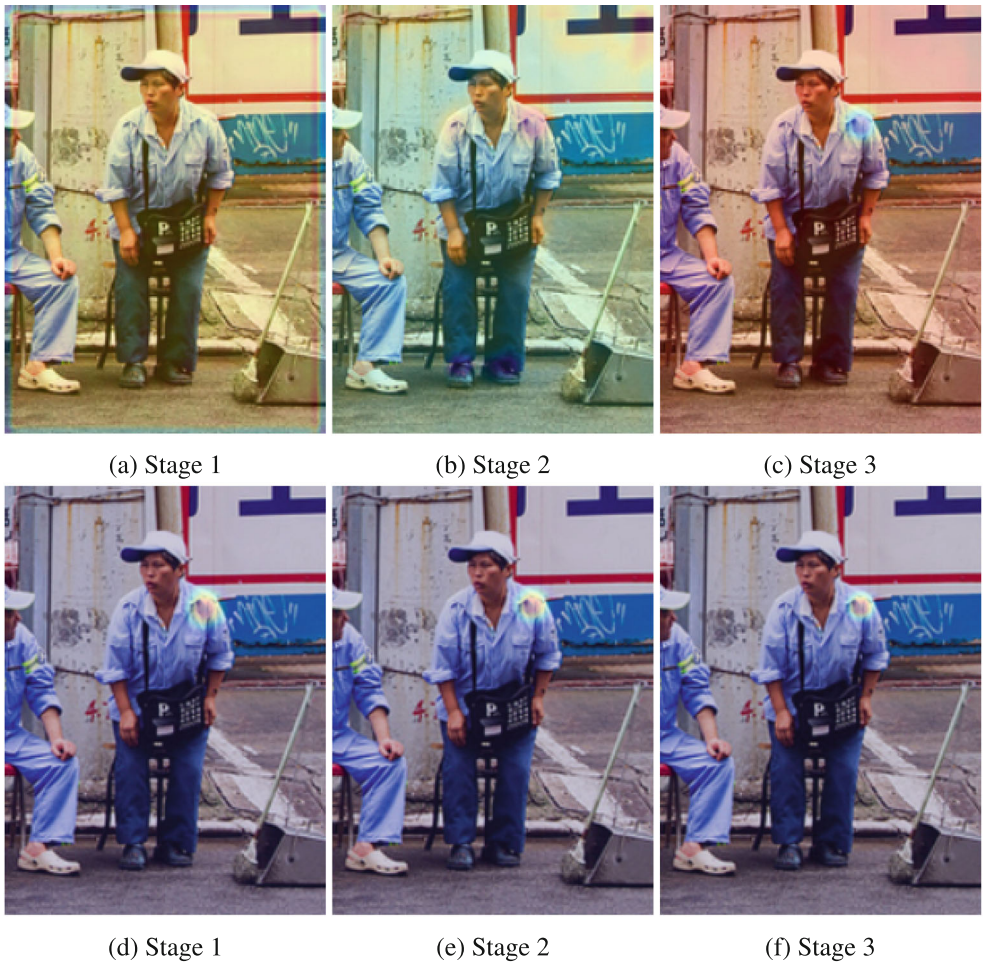
4.6.3 Qualitative results on the CrowdPose test set

We visualize the estimation on the CrowdPose test set in Fig. 4. In each column, the upper estimation is obtained by our model, and the other estimation is obtained by HRNet. We highlight the keypoints that HRNet failed to detect but our model successfully detected in white, and the corresponding skeletons are also highlighted in white for better visualization. We find that the HRNet model failed to detect many occluded keypoints, while our model not only detected those occluded keypoints but also estimated them accurately. This demonstrates that our method can improve the performance in crowded scenes.

4.6.4 Effect of the auxiliary task module

We visualize the outputs of the Auxiliary Task module for the left shoulder in Fig. 5. Figure 5a, b, and c correspond to the output using the features from stages 1, 2, and 3. After mapping the output to the range 0-1 and visualizing it as a heatmap, we combined the heatmap with the origin image to facilitate understanding. Starting from the second stage, the output looks like a heatmap, namely, the output shows a pattern on the location of a keypoint. Moreover, the output of stage 2 is a coarse estimation, where higher values are located around both the left shoulder and the right shoulder. The output of stage 3 has smaller values around the left

Fig. 4 Visualization of pose estimation results on the CrowdPose test set. The first row presents the estimation produced by our models. The second row presents the estimation of HRNet



(a) Stage 1 (b) Stage 2 (c) Stage 3

(d) Stage 1 (e) Stage 2 (f) Stage 3

Fig. 5 Visualization of decision-level information without and with the Auxiliary Task module. The first row presents decision-level information obtained from the model without the Auxiliary Task module. The second row presents decision-level information obtained from the model with the Auxiliary Task module.

shoulder and higher values at the other location, which is likely an inversion of the ground truth heatmap. The above finding holds among different samples. Thus, we refer to this as an internal representation that was found by the model.

After adding the Auxiliary Task module, we visualize those outputs again in Fig. 5d, e, and f. We find that the prior information, namely, the form of the decision-level information, was successfully incorporated into the decision-level information. Combined with the results in Table 5, we conclude that the Auxiliary Task module can provide prior information about the form of the decision-level information and improve the performance.

5 Conclusions

In this work, we focused on decision-level information fusion to boost the performance of human pose estimation. We proposed the View Fusion module for fusing the decision-level information to obtain a comprehensive estimation. The Auxiliary Task module was introduced to bridge the gap between the Encoder and the View Fusion module and to provide prior information about the form of the decision-level information. To guide the model gradually giving more attention to the keypoints that are poorly estimated, we used different strategies for the training of different stages of the model. We evaluated our method on both the new challenging CrowdPose dataset and the widely used COCO dataset. Our experimental results showed that our model achieves the best results on the CrowdPose test set and improves the performance in both crowded and uncrowded scenes. The results of ablation studies showed that our method achieves a great balance between performance and computational cost and can realize improved performance with nearly negligible increments of the numbers of parameters and FLOPs.

Further studies demonstrated that the proposed method is independent of the choice of the backbone and improves the performance of various backbones.

Our future work will include extending this approach to other areas, such as semantic segmentation, and exploring how to fuse information from different stages more effectively.

Appendix A: Impact of σ

We present the evaluation results of LiteHRNet on the CrowdPose test set with various values of σ.

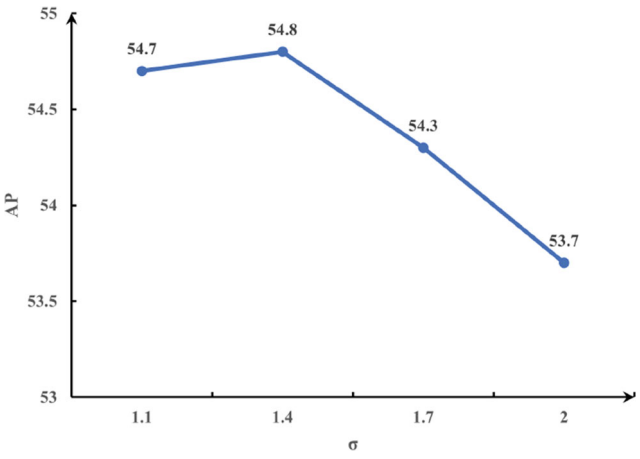


Fig. 6 Impact of σ

As shown in Fig. 6, with the increase of σ, the performance initially increases and subsequently drops. Thus, the choice of σ can affect the performance.

Declarations

Funding This work was supported in part by National Key Research and Development Program of China (No. 2018YFB2101300), in part by National Natural Science Foundation of China (Grant No. 61871186), and in part by the Dean's Fund of Engineering Research Center of Software/Hardware Codesign Technology and Application, Ministry of Education (East China Normal University).

Conflict of Interests The authors have no relevant financial or nonfinancial interests to disclose.

Availability of Data and Material The data that support the findings of this study are openly available. The COCO dataset is available at https://cocodataset.org/. The CrowdPose dataset is available at https://github.com/Jeff-sjtu/CrowdPose.

References

1. Chen Y, Tian Y, He M (2020) Monocular human pose estimation: A survey of deep learning-based methods. Comput Vis Image Underst 192. https://doi.org/10.1016/j.cviu.2019.102897
2. Luvizon D, Picard D, Tabia H (2020) Multi-task Deep Learning for Real-Time 3D Human Pose Estimation and Action Recognition. IEEE Trans Pattern Anal Mach Intell:1–1. https://doi.org/10.1109/TPAMI.2020.2976014
3. Sun Y, Huang H, Yun X, Yang B, Dong K (2021) Triplet attention multiple spacetime-semantic graph convolutional network for skeleton-based action recognition. Appl Intell. https://doi.org/10.1007/s10489-021-02370-x
4. Yoon Y, Yu J, Jeon M (2021) Predictively encoded graph convolutional network for noise-robust skeleton-based action recognition. Appl Intell. https://doi.org/10.1007/s10489-021-02487-z
5. Gao C, Chen Y, Yu J-G, Sang N (2020) Pose-guided spatiotemporal alignment for video-based person Re-identification. Inf Sci 527:176–190. https://doi.org/10.1016/j.ins.2020.04.007
6. Zheng L, Huang Y, Lu H, Yang Y (2019) Pose-Invariant Embedding for Deep Person Re-Identification. IEEE Trans Image Process 28(9):4500–4509. https://doi.org/10.1109/TIP.2019.2910414
7. Liu H, Fang S, Zhang Z, Li D, Lin K, Wang J (2021) MFDNet: Collaborative poses perception and matrix fisher distribution for head pose estimation. IEEE Trans Multimed:1–1. https://doi.org/10.1109/TMM.2021.3081873
8. Li D, Liu H, Zhang Z, Lin K, Fang S, Li Z, Xiong NN (2021) CARM: Confidence-aware recommender model via review representation learning and historical rating behavior in the online platforms. Neurocomputing 455:283–296. https://doi.org/10.1016/j.neucom.2021.03.122
9. Shen X, Yi B, Liu H, Zhang W, Zhang Z, Liu S, Xiong N (2021) Deep Variational Matrix Factorization with Knowledge Embedding for Recommendation System. IEEE Trans Knowl Data Eng 33(5):1906–1918. https://doi.org/10.1109/TKDE.2019.2952849
10. Liu T, Liu H, Li Y, Zhang Z, Liu S (2019) Efficient Blind Signal Reconstruction With Wavelet Transforms Regularization for Educational Robot Infrared Vision Sensing. IEEE/ASME Trans Mechatron 24(1):384–394. https://doi.org/10.1109/TMECH.2018.2870056
11. Liu T, Liu H, Li Y-F, Chen Z, Zhang Z, Liu S (2020) Flexible FTIR Spectral Imaging Enhancement for Industrial Robot Infrared Vision Sensing. IEEE Trans Indust Inform 16(1):544–554. https://doi.org/10.1109/TII.2019.2934728
12. Liu H, Nie H, Zhang Z, Li Y-F (2021) Anisotropic angle distribution learning for head pose estimation and attention understanding in human-computer interaction. Neurocomputing 433:310–322. https://doi.org/10.1016/j.neucom.2020.09.068
13. Li Z, Liu H, Zhang Z, Liu T, Xiong NN (2021) Learning knowledge graph embedding with heterogeneous relation attention networks, IEEE Trans Neural Netw Learn Syst:1–13. https://doi.org/10.1109/TNNLS.2021.3055147
14. Zhang Z, Li Z, Liu H, Xiong NN (2020) Multi-scale dynamic convolutional network for knowledge graph embedding, IEEE Trans Knowl Data Eng:1–1. https://doi.org/10.1109/TKDE.2020.3005952
15. Wei S, Ramakrishna V, Kanade T, Sheikh Y (2016) Convolutional Pose Machines. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 4724–4732
16. Li M, Zhou Z, Liu X (2019) Multi-Person Pose Estimation Using Bounding Box Constraint and LSTM. IEEE Trans Multimed 21(10):2653–2663. https://doi.org/10.1109/TMM.2019.2903455
17. Cheng B, Xiao B, Wang J, Shi H, Huang TS, Zhang L (2020) HigherHRNet: Scale-Aware Representation Learning for Bottom-Up Human Pose Estimation. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp 5385–5394
18. Samet N, Akbas E (2021) HPRNet: Hierarchical point regression for whole-body human pose estimation. Image Vis Comput 115:104285. https://doi.org/10.1016/j.imavis.2021.104285

19. Toshev A, Szegedy C (2014) DeepPose: Human Pose Estimation via Deep Neural Networks. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp 1653–1660
20. Tompson J, Goroshin R, Jain A, LeCun Y, Bregler C (2015) Efficient object localization using Convolutional Networks. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 648–656
21. Newell A, Yang K, Deng J (2016) Stacked hourglass networks for human pose estimation. In: Leibe B, Matas J, Sebe N, Welling M (eds) Computer Vision – ECCV 2016, Lecture Notes in Computer Science. Springer International Publishing, Cham, pp 483–499
22. Xiao B, Wu H, Wei Y (2018) Simple baselines for human pose estimation and tracking. In: Ferrari V, Hebert M, Sminchisescu C, Weiss Y (eds) Computer Vision – ECCV 2018, Lecture Notes in Computer Science. Springer International Publishing, Cham, pp 472–487
23. Tian Y, Hu W, Jiang H, Wu J (2019) Densely connected attentional pyramid residual network for human pose estimation. Neurocomputing 347:13–23. https://doi.org/10.1016/j.neucom.2019.01.104
24. Huang J, Zhu Z, Guo F, Huang G (2020) The devil is in the details: delving into unbiased data processing for human pose estimation. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp 5699–5708
25. Wang J, Sun K, Cheng T, Jiang B, Deng C, Zhao Y, Liu D, Mu Y, Tan M, Wang X, Liu W, Xiao B (2021) Deep high-resolution representation learning for visual recognition. IEEE Trans Pattern Anal Mach Intell 43(10):3349–3364. https://doi.org/10.1109/TPAMI.2020.2983686
26. Chen Y, Wang Z, Peng Y, Zhang Z, Yu G, Sun J (2018) Cascaded pyramid network for multi-person pose estimation. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 7103–7112
27. Cai Y, Wang Z, Luo Z, Yin B, Du A, Wang H, Zhang X, Zhou X, Zhou E, Sun J (2020) Learning delicate local representations for multi-person pose estimation. In: Vedaldi A, Bischof H, Brox T, Frahm J-M (eds) Computer Vision – ECCV 2020. Springer International Publishing, Cham, pp 455–472
28. Yan M, Deng Z, He B, Zou C, Wu J, Zhu Z (2022) Emotion classification with multichannel physiological signals using hybrid feature and adaptive decision fusion. Biomed Signal Process Control 71:103235. https://doi.org/10.1016/j.bspc.2021.103235
29. Liu A-A, Lu Z, Xu N, Nie W, Li W (2021) Multi-type decision fusion network for visual Q&A. Image Vis Comput 115:104281. https://doi.org/10.1016/j.imavis.2021.104281
30. Geng X, Liang Y, Jiao L (2020) Multi-frame decision fusion based on evidential association rule mining for target identification. Appl Soft Comput 94:106460. https://doi.org/10.1016/j.asoc.2020.106460
31. Sun X, Xiao B, Wei F, Liang S, Wei Y (2018) Integral human pose regression. In: Ferrari V, Hebert M, Sminchisescu C, Weiss Y (eds) Computer Vision – ECCV 2018, Lecture Notes in Computer Science. Springer International Publishing, Cham, pp 536–553
32. Papandreou G, Zhu T, Kanazawa N, Toshev A, Tompson J, Bregler C, Murphy K (2017) Towards accurate multi-person pose estimation in the wild. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 3711–3719
33. Zhang W, Fang J, Wang X, Liu W (2021) EfficientPose: Efficient human pose estimation with neural architecture search. Comput Vis Media 7(3):335–347. https://doi.org/10.1007/s41095-021-0214-z
34. Oh S-I, Kang H-B (2017) Object detection and classification by decision-level fusion for intelligent vehicle systems. Sens (Basel, Switzerland) 17(1):207. https://doi.org/10.3390/s17010207
35. Zhang J, Tian J, Cao Y, Yang Y, Xu X (2020) Deep time-frequency representation and progressive decision fusion for ECG classification. Knowl-Based Syst 190:105402. https://doi.org/10.1016/j.knosys.2019.105402
36. Lin T-Y, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL (2014) Microsoft COCO: Common objects in context. In: Fleet D, Pajdla T, Schiele B, Tuytelaars T (eds) Computer Vision–ECCV 2014, Lecture Notes in Computer Science. Springer International Publishing, Cham, pp 740–755
37. Li J, Wang C, Zhu H, Mao Y, Fang H-S, Lu C (2019) CrowdPose: efficient crowded scenes pose estimation and a new benchmark. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp 10855–10864
38. Geng Z, Sun K, Xiao B, Zhang Z, Wang J (2021) Bottom-up human pose estimation via disentangled keypoint regression. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 14676–14686
39. Cao Z, Hidalgo G, Simon T, Wei S-E, Sheikh Y (January 2021) OpenPose: realtime multi-person 2d pose estimation using part affinity fields. IEEE Trans Pattern Anal Mach Intell 43(1):172–186. https://doi.org/10.1109/TPAMI.2019.2929257
40. Xiao J, Li H, Qu G, Fujita H, Cao Y, Zhu J, Huang C (2021) Hope: Heatmap and offset for pose estimation. J Ambient Intell Human Comput. https://doi.org/10.1007/s12652-021-03124-w
41. He K, Gkioxari G, Dollár P, Girshick R (2020) Mask R-CNN. IEEE Trans Pattern Anal Mach Intell 42(2):386–397. https://doi.org/10.1109/TPAMI.2018.2844175
42. Fang H-S, Xie S, Tai Y-W, Lu C (2017) RMPE: Regional Multi-person Pose Estimation. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp 2353–2362
43. Xu X, Zou Q, Lin X (2021) CFENet: Content-aware feature enhancement network for multi-person pose estimation. Appl Intell. https://doi.org/10.1007/s10489-021-02383-6
44. Khirodkar R, Chari V, Agrawal A, Tyagi A (2021) Multi-Instance Pose Networks: Rethinking Top-Down Pose Estimation. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV)
45. Qiu L, Zhang X, Li Y, Li G, Wu X, Xiong Z, Han X, Cui S (2020) Peeking into occluded joints: a novel framework for crowd pose estimation. In: Vedaldi A, Bischof H, Brox T, Frahm J-M (eds) Computer Vision – ECCV 2020, Lecture Notes in Computer Science. Springer International Publishing, Cham, pp 488–504
46. Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, Desmaison A, Kopf A, Yang E, DeVito Z, Raison M, Tejani A, Chilamkurthy S, Steiner B, Fang L, Bai J, Chintala S (2019) PyTorch: An imperative style, high-performance deep learning library. In: Advances in Neural Information Processing Systems. Curran Associates, Inc., pp 8024–8035
47. Kingma DP, Ba J (2015) Adam: A method for stochastic optimization. In: Bengio Y, LeCun Y (eds) International Conference on Learning Representations, San Diego
48. Yu C, Xiao B, Gao C, Yuan L, Zhang L, Sang N, Wang J (2021) Lite-HRNet: a lightweight high-resolution network. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 10440–10450
49. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 770–778