Effective Whole-body Pose Estimation with Two-stages Distillation

Zhendong Yang\textsuperscript{1,2*}  Ailing Zeng\textsuperscript{2}  Chun Yuan\textsuperscript{1†}  Yu Li\textsuperscript{2†}

\textsuperscript{1}Tsinghua Shenzhen International Graduate School
\textsuperscript{2}International Digital Economy Academy (IDEA)

\texttt{yangzd21@mails.tsinghua.edu.cn}
\texttt{yuanc@sz.tsinghua.edu.cn}  \{zengailing, liyu\}@idea.edu.cn

Abstract

Whole-body pose estimation localizes the human body, hand, face, and foot keypoints in an image. This task is challenging due to multi-scale body parts, fine-grained localization for low-resolution regions, and data scarcity. Meanwhile, applying a highly efficient and accurate pose estimator to widely human-centric understanding and generation tasks is urgent. In this work, we present a two-stage pose Distillation for Whole-body Pose estimators, named DWPose, to improve their effectiveness and efficiency. The first-stage distillation designs a weight-decay strategy while utilizing a teacher’s intermediate feature and final logits with both visible and invisible keypoints to supervise the student from scratch. The second stage distills the student model itself to further improve performance. Different from the previous self-knowledge distillation, this stage finetunes the student’s head with only 20% training time as a plug-and-play training strategy. For data limitations, we explore the UBody dataset that contains diverse facial expressions and hand gestures for real-life applications. Comprehensive experiments show the superiority of our proposed simple yet effective methods. We achieve new state-of-the-art performance on COCO-WholeBody, significantly boosting the whole-body AP of RTMPose-l from 64.8% to 66.5%, even surpassing RTMPose-x teacher with 65.3% AP. We release a series of models with different sizes, from tiny to large, for satisfying various downstream tasks. Our code and models are available at \url{https://github.com/IDEA-Research/DWPose}.

1. Introduction

Whole-body pose estimation plays a crucial role in numerous human-centric perception, understanding, and generation tasks, including 3D whole-body mesh recovery \cite{1, 25, 31, 65}, human-object interaction \cite{7, 42}, and pose-conditioned human image and motion generation \cite{10, 24, 27, 59}. Furthermore, capturing human poses for virtual content creation and VR/AR has gained significant popularity, relying on user-friendly algorithms like OpenPose \cite{2} and MediaPipe \cite{29, 62}. Despite the convenience of these tools, their performance remains unsatisfactory, limiting their potential. Therefore, further advancements in human pose estimation technology are essential to fully unleash the potential of user-driven content creation. Compared with human pose estimation with body-only keypoints detection, whole-body pose estimation faces more challenges from 1) the hierarchical structures of the human body for fine-grained keypoints localization; 2) the small resolutions of hand and face; 3) the complex body parts matching for multiple persons in an image, especially for occlusion and complex hand poses; 4) data limitation, especially for diverse hand pose and head pose for the whole-body images.

Besides, before deploying a model, it is essential to compress it into a lightweight network. The basic compression tools comprise distillation \cite{14}, pruning \cite{8}, and quanti-
zation [60]. Knowledge distillation (KD) is proposed to enhance the efficiency of a compact model without incurring extra costs during inference. This technique enables a student to inherit knowledge from a larger teacher and has found widespread application in various tasks, such as classification [69], detection [49], and segmentation [28].

In this paper, we explore KD for whole-body pose estimation to benefit many downstream applications, resulting in a series of real-time pose estimators with high performance and efficiency. Specifically, we propose a novel two-stages pose distillation framework DWPose, which achieves state-of-the-art performance, as shown in Fig. 1. We adopt the latest pose estimator RTMPose [18] as the basic model, which has been trained on COCO-WholeBody [19, 26].

In the first-stage distillation, we natively leverage the teacher’s (e.g., RTMPose-x) intermediate layer and final logits to guide the student model (e.g., RTMPose-l). Previous poses training distinguishes keypoints via visibility and only uses visible keypoints for supervision. Unlike that, we use the teacher’s complete outputs with both visible and invisible keypoints as final logits, which can impart reasonable and comprehensive values to facilitate the student’s learning process. Meanwhile, we employ a weight-decay strategy to enhance the efficacy, gradually reducing the distillation’s weight throughout the entire training phase. Due to a better head will determine a more precise localization, the second-stage distillation proposes a head-aware self-KD to enhance the capacity of the head. We construct two identical models and select one as the teacher and the other as the student to be updated. The student backbone is frozen, and only its head is updated through the logit-based distillation. Notably, this plug-and-play approach allows the student to achieve better results with 20% training time, whether trained from scratch with distillation or without, and can be used for any dense prediction heads.

Data volume and diversity addressing different scales of human body parts will affect the model performance. Suffering from the limited holistic annotated keypoints on existing datasets, existing estimators fail to localize well on fine-grained fingers and face landmarks. Thus, the data impact by incorporating an additional UBoby [25] dataset, primarily comprising diverse face and hand keypoints captured in various real-life scenes.

Therefore, our contributions can be summarized as:

- Based on the latest RTMPose as our base model, our proposed distillation and data strategies can significantly improve RTMPose-l from 64.8% to 66.5% AP, even surpassing RTMPose-x teacher with 65.3% AP. We also validate the powerful effectiveness and efficiency of DWPose on the generation task.

2. Related work

2.1. 2D Whole-body Pose Estimation

This task targets locating expressive body, hand, feet, and face keypoints for all persons in an image simultaneously [2, 13, 19]. Due to the lack of whole-body annotations, most previous models are designed for body-only [22, 40, 47, 53], hand-only [6, 32, 62, 68], or face-only [21, 46, 67]. Openpose [3] combines different datasets for separate body parts. MediaPipe [29, 62] builds a perception pipeline for easy-to-use applications, especially for whole-body landmark detection. With the emergence of whole-body data [9, 19], the models for whole-body pose estimation make great progress [13, 18, 48]. Specifically, ZoomNet [19] proposes the first top-down method with a hierarchical single network to solve the scale variance of different body parts. ZoomNAS [48] further explores a neural architecture search framework for jointly searching the model architecture and the connections between different sub-modules to promote both accuracy and efficiency. TCFormer [61] introduces progressive clustering and merging vision tokens for various locations, sizes, and shapes in multiple stages, preserving different scale information well. Recently, RTMPose [18] has discussed key factors in pose estimation and built a real-time model, achieving state-of-the-art results on COCO-WholeBody. However, it still suffers from redundant model designs and data limitations, especially for diverse hand and face poses.

2.2. Knowledge Distillation

Knowledge distillation is a way to compress the model. Hinton et al. [14] first proposed to supervise the student with the soft labels from the teacher’s output. The method is originally designed for classification and is also called logit-based distillation. Some following works [15, 52, 54] utilize teacher’s logits in different ways, transferring more knowledge from soft labels, target and non-target logits [16, 58, 64, 66, 58]. From the logit-based distillation to feature-based distillation, the knowledge is transferred from intermediate layers [17, 55, 57] and it extends the distillation to various tasks, including detection [4, 56], segmentation [38], generation [30] and so on.

Utilizing KD in human pose estimation has been rarely studied [23, 35, 45, 50]. Existing works either distill the heavy heatmaps for body-only pose estimation [23, 35] or focus on gathering separate body-part experts’ knowledge
3. Method

In the following, we provide a detailed exposition of the two-stage pose distillation (TPD). As shown in Fig. 2, it comprises two distinct stages. The first-stage distillation involves a pre-trained teacher guiding the student from scratch at both the feature and logit levels. On the other hand, the second-stage distillation can be considered a self-KD approach. The model employs its own logits to train its head without any labeled data, leading to significant performance enhancements within a concise training period.

3.1. The First-stage distillation

We denote the feature from the teacher’s and student’s backbone as $F^t$ and $F^s$, and the teacher and student’s final output logit as $T^t_i$ and $S^s_i$. The first-stage distillation forces the student to learn the teacher’s feature $F^t$ and logit $T^t_i$.

3.1.1 Feature-based distillation

For the feature-based distillation, we force the student to mimic the teacher’s layer from the backbone directly. We utilize MSE loss to calculate the distance between the student’s feature $F^s$ and the teacher’s feature $F^t$. To learn the knowledge from the teacher’s feature map, the distillation loss of the feature can be formulated as:

$$L_{fe} = \frac{1}{CHW} \sum_{c=1}^{C} \sum_{h=1}^{H} \sum_{w=1}^{W} \left( F^t_{c,h,w} - f(F^s_{c,h,w}) \right)^2,$$

where $f$ is a $1 \times 1$ convolutional layer to reshape the $F^s$ to the same dimension as $F^t$. $H, W, C$ denote the height, width and channel of the teacher’s feature.

3.1.2 Logit-based distillation

RTMPose [18] predicts pose keypoints with a SimCC-based [22] algorithm that treats keypoint localization as a classification task for horizontal and vertical coordinates. Following this design, we can also apply the logit-based knowledge method to it. To begin with, we review the original classification loss for RTMPose as follows:

$$L_{ori} = - \sum_{n=1}^{N} \sum_{k=1}^{K} W_{n,k} \cdot \sum_{i=1}^{L} \frac{1}{V_i} \log(S_i),$$

where $N$ is the number of the person samples in a batch, $K$ is the number of keypoints, e.g., 133 for COCO-WholeBody [19], $L$ is the length of the x or y localization bins. $W_{n,k}$ is a target weight mask to distinguish invisible keypoints. $V_i$ is the label value.

For the logit-based distillation, we follow the form of the original loss $L_{ori}$. It’s worth noting that we drop the target
weight mask $W$ for distillation. Different from the label value, the invisible keypoints can also be distributed a reasonable value by the teacher. So we argue such value is also helpful, and we also verify it in Sec. 5.6. The distillation loss of the logits can be formulated as:

$$L_{\text{logit}} = -\frac{1}{N} \cdot \sum_{n=1}^{N} \sum_{k=1}^{K} \sum_{l=1}^{L} T_i \log(S_l).$$

### 3.1.3 Weight-decay strategy for distillation

With feature distillation loss $L_{\text{fea}}$ and logits distillation loss $L_{\text{logit}}$, we can train the student with the total loss as:

$$L = L_{\text{ori}} + \alpha L_{\text{fea}} + \beta L_{\text{logit}},$$

where $\alpha$ and $\beta$ are hyper-parameters to balance the loss. Inspired by a detection distillation method TADF [41], we apply a weight-decay strategy for the distillation to reduce the distillation penalty gradually. This strategy helps the student to focus more on the label and achieve better performance. We utilize a time function $r(t)$ to implement the strategy, which is as follows:

$$r(t) = 1 - (t - 1)/t_{\text{max}},$$

where $t \in (1, ..., t_{\text{max}})$ is the current epoch and $t_{\text{max}}$ is the total epochs for training. Then the final loss for the first-stage distillation can be formulated as:

$$L_{s1} = L_{\text{ori}} + r(t) \cdot \alpha L_{\text{fea}} + r(t) \cdot \beta L_{\text{logit}},$$

### 3.2. The Second-stage distillation

In the second distillation stage, we try to utilize the trained student model to teach itself for a better performance. In this way, it can bring improvements for the students, whether trained from scratch with distillation or not.

The pose estimator comprises the encoder (backbone) and decoder (head). Based on the trained model, we first build a student with a trained backbone and an untrained head. The teacher is the same model with a trained backbone and head. During training, we freeze the student’s backbone and update the head. Because the teacher and the student have the same architecture, we only need to extract the feature from the backbone once. Then, the feature is fed into the teacher’s trained head and the student’s untrained head to get the logits $T_i$ and $S_l$, respectively. Following the form in Eq. 3, we train the student with $L_{\text{logit}}$ for the second-stage distillation. It’s worth noting that we drop the original loss $L_{\text{ori}}$, which is calculated with label value. Using $\gamma$ to denote the hyper-parameter for loss scale, the final loss for the second-stage distillation can be formulated as:

$$L_{s2} = \gamma L_{\text{logit}}.$$
<table>
<thead>
<tr>
<th>Method</th>
<th>Input Size</th>
<th>GFLOPs</th>
<th>whole-body</th>
<th>body</th>
<th>foot</th>
<th>face</th>
<th>hand</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>AP</td>
<td>AR</td>
<td>AP</td>
<td>AR</td>
<td>AP</td>
</tr>
<tr>
<td>Whole-body</td>
<td></td>
<td></td>
<td>32.7</td>
<td>45.6</td>
<td>42.7</td>
<td>58.3</td>
<td>9.9</td>
</tr>
<tr>
<td>OpenPose [2]</td>
<td>N/A</td>
<td>N/A</td>
<td>44.2</td>
<td>52.3</td>
<td>56.3</td>
<td>61.2</td>
<td>53.2</td>
</tr>
<tr>
<td>Bottom-up</td>
<td></td>
<td></td>
<td>29.4</td>
<td>40.5</td>
<td>38.1</td>
<td>52.6</td>
<td>5.3</td>
</tr>
<tr>
<td>PAF† [3]</td>
<td>512×512</td>
<td>329.1</td>
<td>29.6</td>
<td>40.5</td>
<td>38.1</td>
<td>52.6</td>
<td>5.3</td>
</tr>
<tr>
<td>AE [34]</td>
<td>512×512</td>
<td>212.4</td>
<td>44.0</td>
<td>54.5</td>
<td>58.0</td>
<td>66.1</td>
<td>57.7</td>
</tr>
<tr>
<td>DeepPose [43]</td>
<td>384×288</td>
<td>17.3</td>
<td>33.5</td>
<td>48.4</td>
<td>44.4</td>
<td>56.8</td>
<td>36.8</td>
</tr>
<tr>
<td>SimpleBaseline [47]</td>
<td>384×288</td>
<td>20.4</td>
<td>57.3</td>
<td>67.1</td>
<td>66.6</td>
<td>74.7</td>
<td>63.5</td>
</tr>
<tr>
<td>HRNet [40]</td>
<td>384×288</td>
<td>16.0</td>
<td>58.6</td>
<td>67.4</td>
<td>70.1</td>
<td>77.3</td>
<td>58.6</td>
</tr>
<tr>
<td>PVT [44]</td>
<td>384×288</td>
<td>19.7</td>
<td>58.9</td>
<td>68.9</td>
<td>67.3</td>
<td>76.1</td>
<td>66.0</td>
</tr>
<tr>
<td>FastPose50-dcn-si [9]</td>
<td>256×192</td>
<td>6.1</td>
<td>59.2</td>
<td>66.5</td>
<td>70.6</td>
<td>75.6</td>
<td>70.2</td>
</tr>
<tr>
<td>ZoomNet [19]</td>
<td>384×288</td>
<td>28.5</td>
<td>63.0</td>
<td>74.2</td>
<td>74.5</td>
<td>81.0</td>
<td>60.9</td>
</tr>
<tr>
<td>ZoomNAS [48]</td>
<td>384×288</td>
<td>18.0</td>
<td>65.4</td>
<td>74.4</td>
<td>74.0</td>
<td>80.7</td>
<td>61.7</td>
</tr>
<tr>
<td>Top-down</td>
<td></td>
<td></td>
<td>122.9</td>
<td>61.2</td>
<td>75.9</td>
<td>77.9</td>
<td>63.3</td>
</tr>
<tr>
<td>ViTPose+-S [51]</td>
<td>256×192</td>
<td>5.4</td>
<td>54.4</td>
<td>-</td>
<td>71.6</td>
<td>-</td>
<td>72.1</td>
</tr>
<tr>
<td>ViTPose+-H [51]</td>
<td>256×192</td>
<td>122.9</td>
<td>61.2</td>
<td>-</td>
<td>75.9</td>
<td>-</td>
<td>77.9</td>
</tr>
<tr>
<td>RTMPose-m</td>
<td>256×192</td>
<td>2.2</td>
<td>58.2</td>
<td>67.4</td>
<td>67.3</td>
<td>75.0</td>
<td>61.5</td>
</tr>
<tr>
<td>RTMPose-l</td>
<td>256×192</td>
<td>4.5</td>
<td>61.1</td>
<td>70.0</td>
<td>69.5</td>
<td>76.9</td>
<td>65.8</td>
</tr>
<tr>
<td>RTMPose-x</td>
<td>384×288</td>
<td>10.1</td>
<td>64.8</td>
<td>73.0</td>
<td>71.2</td>
<td>78.1</td>
<td>69.3</td>
</tr>
<tr>
<td>RTMPose-l + UBody</td>
<td>256×192</td>
<td>4.5</td>
<td>62.1</td>
<td>70.6</td>
<td>69.7</td>
<td>76.9</td>
<td>65.5</td>
</tr>
<tr>
<td>RTMPose-l + UBody</td>
<td>384×288</td>
<td>10.1</td>
<td>65.4</td>
<td>73.2</td>
<td>71.0</td>
<td>77.9</td>
<td>68.6</td>
</tr>
<tr>
<td>DWPose-t</td>
<td>256×192</td>
<td>0.5</td>
<td>48.5</td>
<td>58.4</td>
<td>58.5</td>
<td>67.0</td>
<td>46.5</td>
</tr>
<tr>
<td>DWPose-s</td>
<td>256×192</td>
<td>0.9</td>
<td>53.8</td>
<td>63.2</td>
<td>63.3</td>
<td>71.3</td>
<td>53.3</td>
</tr>
<tr>
<td>DWPose-m</td>
<td>256×192</td>
<td>2.2</td>
<td>60.6</td>
<td>69.5</td>
<td>68.5</td>
<td>76.1</td>
<td>63.6</td>
</tr>
<tr>
<td>DWPose-l</td>
<td>256×192</td>
<td>4.5</td>
<td>63.1</td>
<td>71.7</td>
<td>70.4</td>
<td>77.7</td>
<td>66.2</td>
</tr>
<tr>
<td>DWPose-l</td>
<td>384×288</td>
<td>10.1</td>
<td>66.5</td>
<td>74.3</td>
<td>72.2</td>
<td>78.9</td>
<td>70.4</td>
</tr>
</tbody>
</table>

Table 1. Results of Whole-body pose estimation on COCO-WholeBody [19, 48] V1.0 dataset. The teacher that guides DWPose-l and DWPose-m,s,t is RTMPose-x and RTMPose-l, respectively. “†” indicates multi-scale testing. Flip test is used.

Figure 3. Qualitative comparisons of RTMPose-l (left) and DWPose-l (right). Best viewed in color with zoom-in for small parts.

RTMPose-x with fewer parameters and flops. DWPose-l also achieves the new state-of-the-art model for human whole-body pose estimation. With the proposed distillation TPD and more data, we provide a series of effective models with competitive accuracy.

Fig. 3 shows some qualitative comparisons of how our
Figure 4. Qualitative comparisons with two popular whole-body pose estimators. (a) OpenPose; (b) MediaPipe; (c) Our DWPose-l.

<table>
<thead>
<tr>
<th>Method</th>
<th>RTMPose*-x-l</th>
</tr>
</thead>
<tbody>
<tr>
<td>UBody</td>
<td>✓✓</td>
</tr>
<tr>
<td>TPD</td>
<td>-</td>
</tr>
</tbody>
</table>

| body    | 69.5 | 69.7 | 70.4 |
| foot    | 65.8 | 65.5 | 66.2 |
| face    | 83.3 | 84.1 | 84.3 |
| hand    | 51.9 | 55.1 | 56.6 |
| whole-body | 61.1 | 62.1 | 63.1 |

Table 2. Ablation study of the Two-stages Pose Distillation (TPD) method and the UBody dataset. The teacher and student models used are RTMPose-x and RTMPose-l, respectively.

5. Analysis

5.1. Effects of TPD Method and UBody Data

We compare our state-of-the-art model with two widely used models OpenPose [2] and MediaPipe [29, 62], as presented in Fig. 4. Our DWPose also surpasses the other two methods significantly, especially for the robustness of truncation, occlusion, and effectiveness of fine-grained localization. This enables our method to replace these popular methods to benefit corresponding downstream applications effectively.

5.2. Performance on UBody

We first evaluate our method on the COCO WholeBody dataset, as we describe above. In this subsection, we evaluate the models on the UBody dataset, as shown in Tab. 3. We compare the models under two different input resolutions and report the corresponding AP of different human parts. The extra UBody data for training and our distillation method TPD are both helpful to the students, bringing them significant improvements under both input resolutions. Different from COCO, the gains that our TPD brings on UBody mainly focus on the face and hand. As for COCO, the performance on the body, foot, and hand all get significant improvements, but the gains on the face are limited, as shown in Tab. 2. The results on UBody also demonstrate the effectiveness of our distillation method TPD.

5.3. Effects of First and Second Stage Distillation

We propose the two-stage pose distillation (TPD), which includes the first and second stage distillation. To evaluate the impact of each distillation stage, we conduct experiments by using RTMPose-x to distill RTMPose-l on the mixed dataset, as presented in Tab. 4. Both two distillation stages are beneficial for the students, and their combination leads to further improvements in performance. When combining the first-stage and second-stage distillation together, we achieve 63.1 whole AP, which surpasses the performance achieved by using either distillation loss alone.
Table 4. Ablation study of the two distillation stages. The teacher and student are RTMPose-x and RTMPose-l. '*' denotes the model is trained on COCO + UBody.

<table>
<thead>
<tr>
<th>Method</th>
<th>RTMPose* x-l</th>
</tr>
</thead>
<tbody>
<tr>
<td>First-stage</td>
<td>✓ - ✓ - ✓</td>
</tr>
<tr>
<td>Second-stage</td>
<td>✓ ✓</td>
</tr>
<tr>
<td>body</td>
<td>69.7 70.4 69.7 70.4</td>
</tr>
<tr>
<td>foot</td>
<td>65.5 65.8 65.9 66.2</td>
</tr>
<tr>
<td>face</td>
<td>84.1 84.1 84.2 84.3</td>
</tr>
<tr>
<td>hand</td>
<td>55.1 56.4 55.4 56.6</td>
</tr>
<tr>
<td>whole-body</td>
<td>62.1 62.9 62.2 63.1</td>
</tr>
</tbody>
</table>

Table 5. The impact of the proposed head-aware self-KD in the second-stage distillation (S2) on existing estimator RTMPose. '*' denotes the model is trained on COCO + UBody. All results are reported with AP on COCO-WholeBody.

<table>
<thead>
<tr>
<th>Method</th>
<th>body</th>
<th>foot</th>
<th>face</th>
<th>hand</th>
<th>whole-body</th>
</tr>
</thead>
<tbody>
<tr>
<td>RTMPose-m</td>
<td>69.1</td>
<td>64.8</td>
<td>81.8</td>
<td>49.8</td>
<td>60.0</td>
</tr>
<tr>
<td>RTMPose-m + S2</td>
<td>69.4</td>
<td>65.1</td>
<td>81.9</td>
<td>50.3</td>
<td>60.4</td>
</tr>
<tr>
<td>RTMPose-l</td>
<td>69.5</td>
<td>65.8</td>
<td>83.3</td>
<td>51.9</td>
<td>61.1</td>
</tr>
<tr>
<td>RTMPose-l + S2</td>
<td>69.6</td>
<td>66.1</td>
<td>83.2</td>
<td>52.3</td>
<td>61.3</td>
</tr>
<tr>
<td>RTMPose-m*</td>
<td>68.6</td>
<td>63.6</td>
<td>82.5</td>
<td>52.3</td>
<td>60.4</td>
</tr>
<tr>
<td>RTMPose-m* + S2</td>
<td>68.5</td>
<td>63.6</td>
<td>82.8</td>
<td>52.7</td>
<td>60.6</td>
</tr>
<tr>
<td>RTMPose-x*</td>
<td>70.3</td>
<td>65.3</td>
<td>84.9</td>
<td>56.4</td>
<td>63.0</td>
</tr>
<tr>
<td>RTMPose-x* + S2</td>
<td>70.4</td>
<td>65.3</td>
<td>84.9</td>
<td>56.6</td>
<td>63.2</td>
</tr>
</tbody>
</table>

It’s worth noting that the second-stage distillation just needs to fine-tune the head, which helps to save much training time. Interestingly, it helps the student to surpass the teacher RTMPose-x with 63.0% AP.

### 5.4. Second-stage Distillation for Trained Models

Our second-stage distillation is available not only for the models trained with our first-stage distillation but also for those trained without distillation. So it can be applied when there lacks a better and larger teacher. We can utilize the model itself as a teacher to improve it with a short training time. As shown in Tab. 5, we pick three different models and evaluate our second-stage distillation on COCO and the combination of COCO and UBody. For all settings, models with S2 significantly improve, especially for the foot and hand. Compared with traditional distillation and self-KD, it saves much time in training the model from scratch and costs to obtain a better model.

### 5.5. Ablation Study of the First-stage Distillation

As we describe in Eq. 6, our first-stage distillation calculates the loss through the ground-truth label (GT), teacher’s feature (Fea), and teacher’s logits (Logit). Furthermore, we apply a weight-decay strategy (Decay) to further improve the student. In this subsection, we analyze the effects of every component by using RTMPose-l to distill RTMPose-m, as shown in Tab. 6. The knowledge from the feature brings the student 1.4% AP gains. When combing the distillation on the logit, the AP gains get to 1.6%. This proves that the knowledge from the feature and logit are both helpful and complementary to each other. Finally, the weight-decay strategy brings another 0.3% AP gains, helping the student to achieve 62.3% AP.

Interestingly, we try to drop the GT label and train the student just with the teacher’s logit. The student achieves 60.9% AP, which is even 0.5% higher than the model trained with the GT label. This indicates we can label the new data through a teacher model instead of annotating manually, which can save much cost in time and manual efforts, and achieve a better model through such data for training. However, when combining the feature distillation together, the performance with the teacher’s logit gets lower than that with the GT label. Thus, we adopt the GT, Fea, and Logit together for distillation.

<table>
<thead>
<tr>
<th>GT</th>
<th>Fea</th>
<th>Logit</th>
<th>Decay</th>
<th>whole-body</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓</td>
<td>-</td>
<td>✓</td>
<td>-</td>
<td>60.4</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>-</td>
<td>-</td>
<td>61.8</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>-</td>
<td>60.9</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>62.0</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>62.3</td>
</tr>
</tbody>
</table>

Table 6. Ablation study of the components of first-stage distillation. The teacher and student are RTMPose-l and RTMPose-m. The performance is the whole-body AP on COCO with GT boxes.

<table>
<thead>
<tr>
<th>Logit</th>
<th>Mask</th>
<th>whole-body</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓</td>
<td>-</td>
<td>60.9</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>59.8</td>
</tr>
</tbody>
</table>

Table 7. Ablation study of the target weight mask. The teacher and student is RTMPose-l and RTMPose-m. The performance is the whole-body AP on COCO with GT boxes.

### 5.6. Target Mask for Logit-based Distillation

In our logit-based distillation, we deliberately omit the target weight mask $W$, which is employed to differentiate between visible and invisible keypoints, as shown in Eq. 3. We conducted an in-depth investigation into how this target mask affects the distillation process. As indicated in Tab. 7, it is evident that the presence of the target weight mask significantly hampers the distillation performance, resulting in a notable 1.1% drop in the student’s performance. These results underscore the significance of the teacher’s input for invisible keypoints, affirming its positive impact on the student’s learning process.
5.7. Better Pose, Better Image Generation

Recently, controllable image generation [12, 37, 39, 63, 33, 20] has witnessed significant advancements. For human image generation, precise skeleton information is crucial to guide the pose, particularly for whole-body skeletons. Mainstream techniques like ControlNet [63] often rely on OpenPose [2] due to its efficiency and user-friendly nature in generating human poses. However, OpenPose’s performance, as shown in Tab. 1, reaches only 44.2% AP, which leaves room for improvement. Consequently, we aim to replace OpenPose with our DWPose to enhance ControlNet’s image generation without the need for additional training. Utilizing a top-down approach, we first employ YOLOX [11] to detect all individuals and then use our pose estimator to extract keypoints from the detection results, thus boosting the overall image generation process.

In Fig. 5, we employ ControlNet to visualize and compare the generated images using both OpenPose and our DWPose, demonstrating that a more precise and expressive skeleton leads to higher-quality image generation. Additionally, we present a comparison of inference speed with OpenPose in Tab. 8. Thanks to the efficient architecture of RTMPose [18], DWPose requires only about one percent of the time taken by OpenPose to infer the same image. Moreover, as the number of persons in the image increases, the runtime for OpenPose significantly increases. For a single person, the inference times for OpenPose and DWPose are 5.78 s and 0.068 s, respectively. However, when the number of persons reaches nine, the inference time for OpenPose triples, whereas the inference time for DWPose is only about 1.5 times longer.

6. Conclusion

In this paper, we aim to obtain both an efficient and effective model for human whole-body pose estimation. To this end, we apply distillation to the latest effective RTMPose. Accordingly, we first propose a Two-stage Pose Distillation to enhance the lightweight model’s performance. Moreover, the second-stage distillation is available when a larger teacher lacks, and it only needs a short training time to obtain a better model. Then, we investigate the UBody dataset to further improve its performance, obtaining DWPose. Extensive experiments prove that our method is simple yet effective. We also explore the impact of a better pose estimator on the controllable image generation task.

Acknowledgement. This work was supported by the National Key R&D Program of China (2022YFB4701400/4701402), the SZSTC project Grant (JCYJ2019080917201639, WDZC20200820200655001), Shenzhen Key Laboratory (ZDSYS20210623092001004).
References


[26] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence


