A Ablation study of Curriculum Learning

CL	GFLOPs	Body	Foot	Face	Hand	Whole
✗	**1.99**	66.9	64.9	80.8	42.9	56.5
✓	**1.99**	**71.0**	**70.4**	**81.0**	**45.4**	**59.6(+3.1)**

Table 7: Ablation study of curriculum learning.

To ease the challenge of learning whole-body pose estimation, we suggest implementing curriculum learning. This approach enables the model to gradually tackle increasingly complex tasks, starting with the localization of a few uniform keypoints before progressing to localizing numerous dense keypoints. To assess the necessity and effectiveness of curriculum learning in our model, we conduct ablation experiments. From Tab. 7, the results highlight the significant improvement in accuracy for parts with fewer keypoints, such as body and feet, boosting their AP by 4.1 and 5.5. The reason is that if we directly learn the whole body keypoints, parts with more keypoints, such as face and hands, may dominate the optimization process, leading the model to fall into a local optimum. Consequently, the optimization of the parts with fewer keypoints is under-optimized.

B Visualizations

B.1 Pruning

To confirm the redundancy of the pruned visual tokens, we visualize the pruning results by displaying the original images alongside the retained visual tokens for each group at each stage. Fig. 8 demonstrates that the selected regions for each group become more refined as the network goes deeper. As anticipated, each group can focus on the visual tokens that correspond to its keypoints. Notably, even with a high pruning rate, the facial regions, which have numerous keypoints but smaller areas, are appropriately retained. And our method still works well in complex situations such as crowded scenes (Fig. 8(b)), different body poses(Fig. 8(c)), man-object interaction(Fig. 8(d)), occlusion(Fig. 8(e)), partial bodies(Fig. 8(f)), etc. In crowded scenes, as shown in Fig. 8(b), we can accurately prune out the currently estimated figure. When occluded, as shown in Fig. 8(e), our pruning preserves the unoccluded visual tokens within the group as much as possible. When there is only a partial body, as shown in Fig. 8(f), because of the grouping, our method can prune out the visual tokens of the corresponding region of the corresponding character contained within the current image.

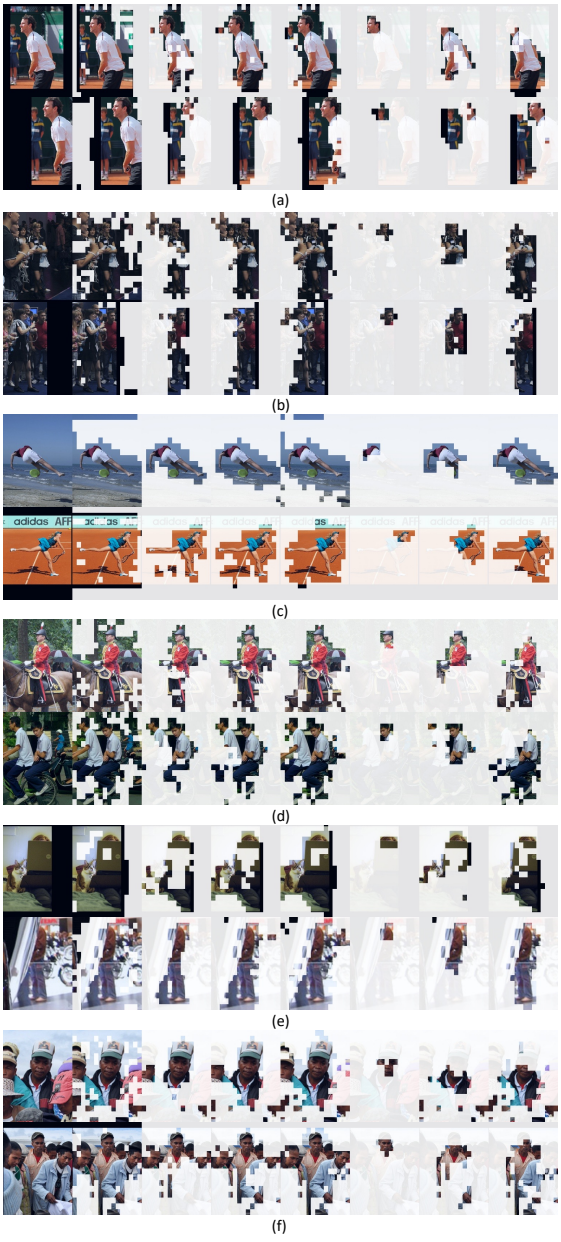

Fig. 8: Visualization of pruning results on COCO-WholeBody validation set. In each case, the first column shows the original image. The second column displays the result of the first pruning. The third to fifth columns represent the result of the second pruning, with the third, fourth, and fifth columns representing the three groups of the head, the upper body, and the lower body. The sixth to eighth columns represent the result of the third pruning, with the sixth, seventh, and eighth columns representing the three groups of the head, the upper body, and the lower body.

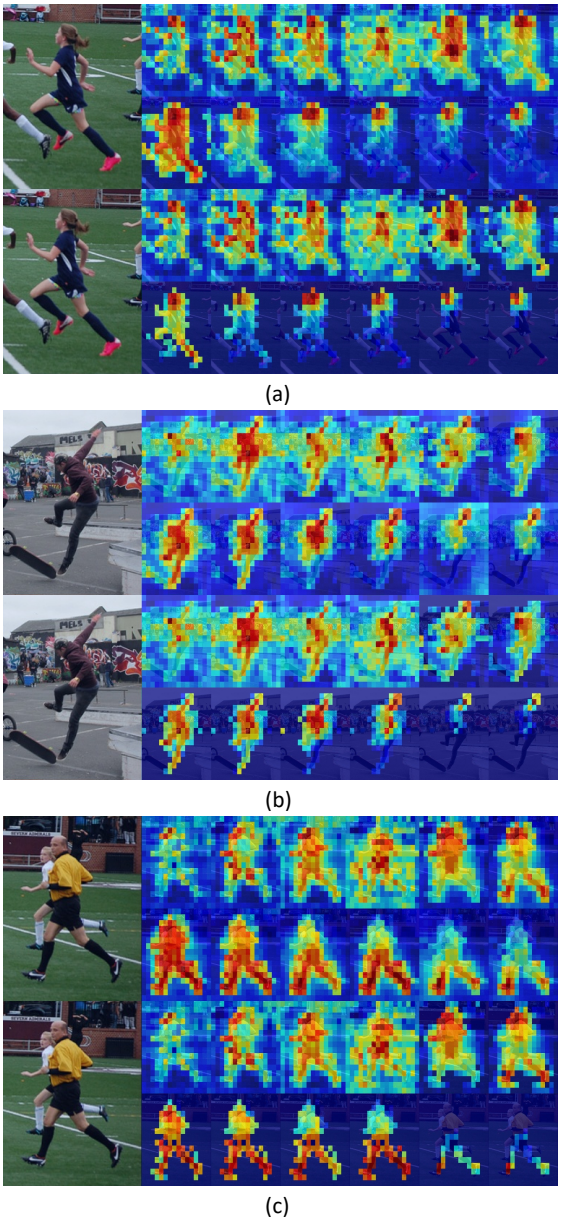

Fig. 9: Visualization of attention maps in different layers on COCO-Wholebody validation set. In each case, the 1^{st} row displays the attention maps for the 1^{st} through 6^{th} layers of the unpruned model. The 2^{nd} row displays the attention maps for the 7^{th} through 12^{th} layers of the unpruned model. The 3^{rd} row displays the attention maps for the 1^{st} through 6^{th} layers of the pruned model. The 4^{th} row displays the attention maps for the 7^{th} through 12^{th} layers of the pruned model. (a) shows the attention maps of the nose. (b) shows the attention maps of the wrist. (c) shows the attention maps of the ankle.

B.2 Attention Maps in different layers

To further validate that our group-based pruning eliminates redundant visual tokens, we visualize the attention maps of selected keypoints before and after pruning in different layers. From the Fig. 9, it is evident that during the initial stages of modeling, regardless of the keypoint, the attention is focused on the entire character's body rather than specific local keypoints. As the model deepens, the keypoints' attention gradually narrows down, focusing solely on the area surrounding each keypoint. Furthermore, we can observe that the pruned visual tokens are indeed redundant. The attention of keypoints focuses on mostly the same visual tokens before and after pruning.

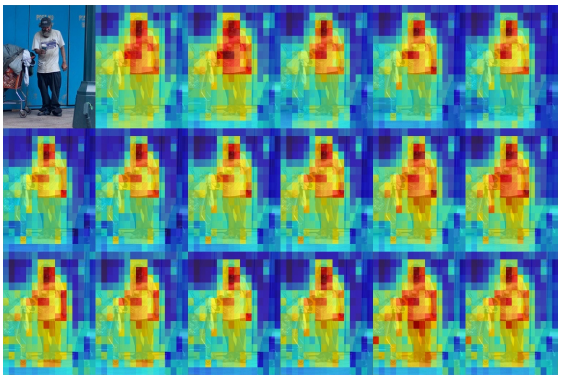

Fig. 10: Visualization of attention maps in the shallow(4^{th}) layer on COCO-Wholebody validation set. Starting from the top left corner, 18 images are shown in order from left to right and top to bottom: original image, nose, left eye, right eye, left ear, right ear, left shoulder, right shoulder, left elbow, right elbow, left wrist, right wrist, left hip, right hip, left knee, right knee, left ankle, and right ankle.

B.3 Attention Maps in the shallow layer

To verify that the regions attended by different keypoints are similar in the shallow layer of the model, we visualized the attention maps of all body keypoints in the shallow layer. We uniformly chose the attention maps of the fourth layer for visualization. The visualization results are shown in Figure 10. As can be seen from the figure, in the shallow layer, all the keypoints focus on the human body, especially on the face. Therefore, the regions that different keypoints focus on are similar in the shallow layer.

Model	Prune	GFLOPs	FPS	Throughput	Memory
GTPT-T	✗	1.33	93	928	1711M
GTPT-T	✓	0.83(**-46%**)	91	1775(+**91%**)	700M(**-59%**)
GTPT-S	✗	3.53	93	701	1864M
GTPT-S	✓	1.99(**-44%**)	91	1231(+**76%**)	784M(**-58%**)
GTPT-B	✗	5.13	63	572	1886M
GTPT-B	✓	4.02(**-22%**)	62	879(+**54%**)	838M(**-56%**)

Table 8: Inference efficiency comparison on COCO-WholeBody validation. The input size is 256×192.

C Inference Efficiency

While GFLOPs already reflect the network's efficiency, it does not directly correspond to the actual runtime on the hardware. It is crucial to measure its actual runtime on the hardware to validate the efficiency of our model. The most commonly used metric for this purpose is FPS (Frames Per Second), which processes only one instance at a time. However, our approach follows a top-down framework. It means that by giving an input image, we first detect all instances of people through a detector, then crop, resize, and combine each human instance. Finally, we feed them into the pose estimation model using a mini-batch to accelerate inference. Therefore, we believe that evaluating the operational efficiency of the model in terms of throughput is more reasonable than using FPS. Throughput measures the maximum number of input instances that can be processed per unit of time, making it more consistent with the actual scenario when processing multiple instances in parallel. In addition, memory usage is also an indicator that we have to care about. On a graphics card with fixed memory, only the model with lower memory usage can handle more different human instances at the same time.

We conducted experiments by setting the batch size to 32, using pytorch, and performing inference on a single A100 GPU to control the variables. The results, including FPS, throughput, and memory usage, are presented in the Tab. 8. Pruning does not reduce the inference time for a single instance. Because we need to sort the visual tokens based on their importance, select those with higher importance, and prune the ones with lower importance during the pruning process. The runtime saved by pruning is then utilized to compensate for the runtime consumed by the sorting process. However, pruning enhances the throughput and reduces its memory usage effectively. As a result, it improves the overall computational efficiency in practical applications. Besides, it is worth noting that our GTPT implementation utilizes PyTorch and does not incorporate efficiency-boosting technologies like FlashAttention [6] that accelerate Transformer inference. It indicates that there is potential for further advancements in the efficiency of GTPT.