PPT: token-Pruned Pose Transformer for monocular and multi-view human pose estimation

Haoyu Ma[1], Zhe Wang[1], Yifei Chen[2], Deying Kong[1], Liangjian Chen[3], Xingwei Liu[1], Xiangyi Yan[1], Hao Tang[3], Xiaohui Xie[1]

[1] University of California, Irvine
{haoyum3, zwang15, deyingk, xingweil, xiangyy4, xhx}@uci.edu
[2] Tencent Inc
dolphinchen@tencent.com
[3] Meta Reality Lab, Meta AI
{clj, haotang}@fb.com

Abstract. Recently, the vision transformer and its variants have played an increasingly important role in both monocular and multi-view human pose estimation. Considering image patches as tokens, transformers can model the global dependencies within the entire image or across images from other views. However, global attention is computationally expensive. As a consequence, it is difficult to scale up these transformer-based methods to high-resolution features and many views.

In this paper, we propose the token-Pruned Pose Transformer (PPT) for 2D human pose estimation, which can locate a rough human mask and performs self-attention only within selected tokens. Furthermore, we extend our PPT to multi-view human pose estimation. Built upon PPT, we propose a new cross-view fusion strategy, called human area fusion, which considers all human foreground pixels as corresponding candidates. Experimental results on COCO and MPII demonstrate that our PPT can match the accuracy of previous pose transformer methods while reducing the computation. Moreover, experiments on Human 3.6M and Ski-Pose demonstrate that our Multi-view PPT can efficiently fuse cues from multiple views and achieve new state-of-the-art results. Source code and trained model can be found at https://github.com/HowieMa/PPT.

Keywords: vision transformer, token pruning, human pose estimation, multi-view pose estimation

1 Introduction

Human pose estimation aims to localize anatomical keypoints from images. It serves as a foundation for many down-stream tasks such as AR/VR, action recognition [21, 65], and medical diagnosis [11]. Over the past decades, deep convolutional neural networks (CNNs) play a dominant role in human pose estimation tasks [53, 62, 40, 63, 50, 59, 61]. However, cases including occlusions and oblique

● query token   ● reference token

(a) Global fusion   (b) Epipolar-based fusion   (c) Human area fusion (ours)
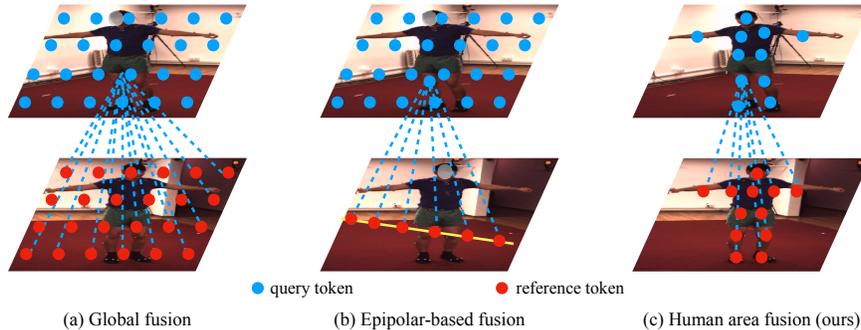
Fig. 1. Different types of cross-view fusion. The first row is the current view, and the second row is the reference view.

viewing are still too difficult to be solved from a monocular image. To this end, some works apply a multi-camera setup [48, 60, 22, 6] to boost the performance of 2D pose detection[43, 19], since difficult cases in one view are potentially easier to be resolved in other views. Meanwhile, human body joints are highly correlated, constrained by strong kinetic and physical constraints [52]. However,since the reception fields of CNNs are limited, the long-range constraints among joints are often poorly captured [31].

Recently, the ViT [14] demonstrates that the transformers [55] can achieve impressive performance on many vision tasks [54, 2]. Compared with CNN, the self-attention module of transformers can easily model the global dependencies among all visual elements. In the field of pose estimation, many tansformer-based works [31, 67, 37, 33, 74] suggest that the global attention is necessary. In single-view 2D human pose estimation, TransPose [67] and TokenPose [31] achieve new state-of-the-art performance and learn the relationship among keypoints with transformers. In multi-view human pose estimation, the TransFusion [36] uses the transformer to fuse cues from both current and reference views. Typically, these works flatten the feature maps into 1D token sequences, which are then fed into the transformer. In multi-view settings, tokens from all views are usually concatenated together to yield a long sequence. However, the dense global attention of transformers is computationally extensive. As a result, it is challenging to scale up these methods to high-resolution feature maps and many views. For example, the TransFusion [36] can only compute global attention between two views due to the large memory cost. Meanwhile, as empirically shown in Fig.2, the attention map of keypoints is very sparse, which only focuses on the body or the joint area. This is because the constraints among human keypoints tend to be adjacent and symmetric [31]. This observation also suggests that the dense attention among all locations in the image is relatively extravagant.

In this paper, we propose a compromised and yet efficient alternative to the global attention in pose estimation, named token-Pruned Pose Transformer (PPT). We calculate attention only within the human body area, rather than over the entire input image. Specifically, we select human body tokens and prune
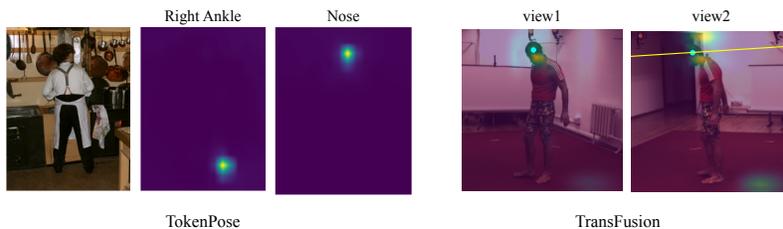
Fig. 2. Attention map for TokenPose (monocular view) and TransFusion (multi-view). The attention maps are very sparse and only attend to a small local regions.

background tokens with the help of attention maps. As the human body only takes a small area of the entire image, the majority of input tokens can be pruned. We reveal that pruning these less informative tokens does not hurt the pose estimation accuracy, but can accelerate the entire networks. Interestingly, as a by-product, PPT can also predict a rough human mask without the guidance of ground truth mask annotations.

Moreover, we extend PPT to multi-view settings. As in Fig.1, previous cross-view fusion methods consider all pixels in the reference view (global fusion) or pixels along the epipolar line (epipolar-based fusion) as candidates. The former is computationally extensive and inevitably introduces noise from the background, and the latter requires accurate calibration and lacks semantic information. Built upon PPT, we propose a new fusion strategy, called *human area fusion*, which considers human foreground pixels as corresponding candidates. Specifically, we firstly use PPT to locate the human body tokens on each view, and then perform the multi-view fusion among these selected tokens with transformers. Thus, our method is an efficient fusion strategy and can easily be extended to many views.

Our main contributions are summarized as follows:

1. We propose the token-Pruned Pose Transformer (PPT) for efficient 2D human pose estimation, which can locate the human body area and prune background tokens with the help of a Human Token Identification module.
2. We propose the strategy of "Human area fusion" for multi-view pose estimation. Built upon PPT, the multi-view PPT can efficiently fuse cues from human areas of multiple views.
3. Experimental results on COCO and MPII demonstrate that our PPT can maintain the pose estimation accuracy while significantly reduce the computational cost. Results on Human 3.6M and Ski-Pose show that human area fusion outperforms previous fusion methods on 2D and 3D metrics.

2   Related Work

2.1   Efficient Vision Transformers

Recently, the transformer [55] achieves great progresses on many computer vision tasks, such as classification [14, 54], object detection [2, 76, 15], and semantic

segmentation [75, 58, 66, 68]. While being promising in accuracy, the vanilla ViT [14] is cumbersome and computationally intensive. Therefore, many algorithms have been proposed to improve the efficiency of vision transformers. Recent works demonstrate that some popular model compression methods such as network pruning [17, 7, 8, 70], knowledge distillation [20, 54, 9], and quantization [46, 51] can be applied to ViTs. Besides, other methods introduce CNN properties such as hierarchy and locality into the transformers to alleviate the burden of computing global attention [35, 5]. On the other hand, some works accelerate the model by slimming the input tokens [71, 3, 45, 44, 29, 32, 38]. Specifically, the Token-to-tokens [71] aims to reduce the number of tokens by aggregating neighboring tokens into one token. The TokenLearner [45] mines important tokens by learnable attention weights conditioned on the input feature. The DynamicViT [44] prunes less informative tokens with an extra learned token selector. The EViT [32] reduces and reorganizes image tokens based on the classification token. However, all these models have only been designed for classification, where the final prediction only depends on the special classification token.

2.2   Human Pose Estimation

Monocular 2D Pose Estimation In the past few years, many successful CNNs are proposed in 2D human pose estimation. They usually capture both low-level and high-level representations [62, 12, 40, 13, 63, 50], or use the structural of skeletons to capture the spatial constraints among joints [52, 24, 42, 26, 27, 10, 28]. Recently, many works introduce transformers into pose estimation tasks [67, 31, 37, 30, 33, 74]. Specifically, TransPose [67] utilizes transformers to explain dependencies of keypoint predictions. TokenPose [31] applies additional keypoint tokens to learn constraint relationships and appearance cues. Both works demonstrate the necessity of global attention in pose estimation.

Efficient 2D Pose Estimation Some recent works also explore efficient architecture design for real-time pose estimation [41, 39, 47, 57, 72, 69]. For example, EfficientPose [72] designs an efficient backbone with neural architecture search. Lite-HRNet [69] proposes the conditional channel weighting unit to replace the heavy shuffle blocks of HRNet. However, these works all focus on CNN-based networks, and none of them study transformer-based networks.

Multi-view Pose Estimation 3D pose estimation from multiple views usually takes two steps: predicting 2D joints on each view separately with a 2D pose detector, and lifting 2D joints to 3D space via triangulation. Recently, many methods focus on enabling the 2D pose detector to fuse information from other views [43, 73, 64, 19, 36]. They can be categorized into two groups: 1) Epipolar-based fusion. The features of one pixel in one view is augmented by fusing features along the corresponding epipolar line of other views. Specifically, the AdaFuse [73] adds the largest response on the heatmap along the epipolar line. The epipolar transformer [19] applies the non-local module [56] on intermediate
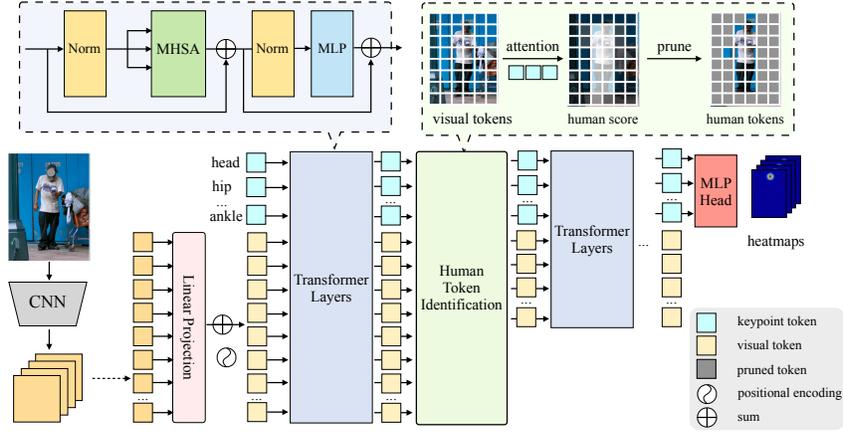
Fig. 3. Framework of the token-Pruned Pose Transformer (PPT). The visual tokens are obtained from the flattened CNN feature maps. The keypoint tokens are added to represent each joint and predict the keypoints heatmaps. The Human Token Identification (HTI) module is inserted inside the transformer layers to locate human visual tokens and prune background tokens. Thus the followed transformer layers are only performed on these selected tokens.

features to obtain the fusion weights. However, this fusion strategy requires precise camera calibration and discard information outside the epipolar lines. 2) Global fusion. The features of one pixel in one view are augmented by fusing features of all locations in other views. In detail, the Cross-view Fusion [43] learns a fixed attention matrix to fuse heatmaps in all other views. The TransFusion [36] applies the transformers to fuse features of the reference views and demonstrates that global attention is necessary. However, the computation complexity of global fusion is quadratic to the resolution of input images and number of views. Thus, both categories have their limitations. A fusion algorithm that can overcome these drawbacks and maintains their advantages is in need.

3   Methodology

3.1   Token-Pruned Pose Transformer

Overview Fig.3 is an overview of our token-Pruned Pose Transformer. Following [31], the input RGB image $\mathbf{I}$ first go through a shallow CNN backbone $\mathcal{B}(\cdot)$ to obtain the feature map $\mathbf{F} \in \mathbb{R}^{H \times W \times C}$. Then $\mathbf{F}$ is decomposed into flattened image patches $\mathbf{F}_p \in \mathbb{R}^{N_v \times (C \cdot P_h \cdot P_w)}$, where (P_h, P_w) is the resolution of each image patch, and $N_v = \frac{H}{P_h} \cdot \frac{W}{P_w}$ is the total number of patches [14]. Then a linear projection is applied to project $\mathbf{F}_p$ into $\mathbf{X}_p \in \mathbb{R}^{N_v \times D}$, where D is the dimension of hidden embeddings. The 2D positional encodings $\mathbf{E} \in \mathbb{R}^{N_v \times D}$ are added to make the transformer aware of position information [55], *i.e.*, $\mathbf{X}_v = \mathbf{X}_p + \mathbf{E}$, namely the visual token. Meanwhile, following TokenPose [31], we have J additional

learnable keypoint tokens $\mathbf{X}_k \in \mathbb{R}^{J \times D}$ to represent J target keypoints. The input sequence to the transformer is $\mathbf{X}^0 = [\mathbf{X}_k, \mathbf{X}_v] \in \mathbb{R}^{N \times D}$, where $N = N_v + J$ and $[\ldots]$ is the concatenation operation.

The transformer has L encoder layers in total. At the L_1^{th} layer, the Human Token Identification (HTI) module locates K most informative visual tokens where human body appears and prunes the remaining tokens. We denote $r = \frac{K}{N_v}(0 < r < 1)$ as the keep ratio. As a result, the length of the sequence is reduced to $N' = rN_v + J$ for the following transformer layers. The HTI is conducted e times at the $L_1^{th}, L_2^{th}, \ldots, L_e^{th}$ layers. Thus, PPT can progressively reduce the length of visual tokens. Finally, the total number of tokens is $r^e N_v + J$. The prediction head projects the keypoint tokens in the last layer $\mathbf{X}_k^L \in \mathbb{R}^{J \times D}$ into the output heatmaps $\mathbf{H} \in \mathbb{R}^{J \times (H_h \cdot W_h)}$.

Transformer Encoder Layer. The encoder layer consists of the multi-headed self-attention (MHSA) and multi-layer perceptron (MLP). Operations in one encoder layer is shown in Fig. 3. The self-attention aims to match a query and a set of key-value pairs to an output [55]. Given the input $\mathbf{X}$, three linear projections are applied to transfer $\mathbf{X}$ into three matrices of equal size, namely the query $\mathbf{Q}$, the key $\mathbf{K}$, and the value $\mathbf{V}$. The self-attention (SA) operation is calculated by:

$$\text{SA}(\mathbf{X}) = \text{Softmax}(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{D}})\mathbf{V}, \tag{1}$$

For MHSA, H self-attention modules are applied to $\mathbf{X}$ separately, and each of them produces an output sequence.

Human Token Identification (HTI). The TokenPose [31] conducts self-attention among all visual tokens, which is cumbersome and inefficient. From Equation 1, we know that each keypoint token $\mathbf{X}_k^j$ interacts with all visual tokens $\mathbf{X}_v$ via the attention mechanism:

$$\text{Softmax}(\frac{\mathbf{q}_k^j \mathbf{K}_v^T}{\sqrt{D}})\mathbf{V}_v = \mathbf{a}^j \mathbf{V}_v, \tag{2}$$

where $\mathbf{q}_k^j$ denotes the query vector of $\mathbf{X}_k^j$, $\mathbf{K}_v$ and $\mathbf{V}_v$ are the keys and values of visual tokens $\mathbf{X}_v$. To this end, each keypoint token is a linear combination of all value vectors of visual tokens. The combination coefficients $\mathbf{a}^j \in \mathbb{R}^{N_v}$ are the attention values from the query vector for that keypoint token with respect to all visual tokens. To put it differently, the attention value determines how much information of each visual token is fused into the output. Thus, it is natural to assume that the attention value $\mathbf{a}^j$ indicates the importance of each visual token in the keypoint prediction [32]. Typically, a large attention value suggests that the target joint is inside or nearby the corresponded visual token.

With this assumption, we propose the Human Token Identification module to select informative visual tokens with the help of attention scores of keypoint
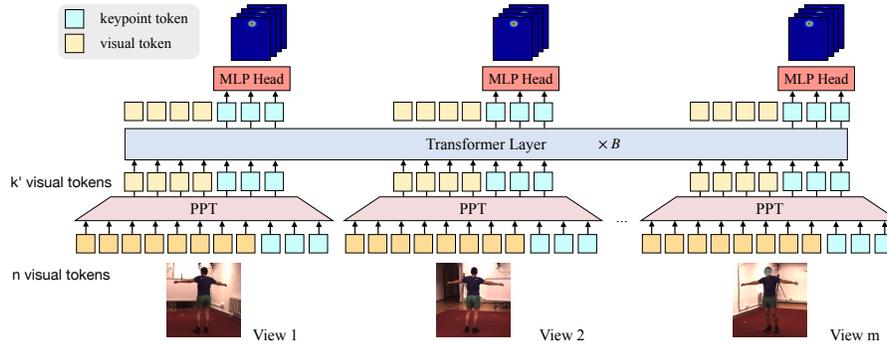
Fig. 4. Overall framework of the Multi-view PPT. A share-weight PPT is applied to extract a subset of visual tokens for each view. Then B transformer layers are applied to the concatenated tokens from each view to perform cross-view fusion. The output head takes keypoint tokens in each view to predict heatmaps.

tokens. However, each keypoint token usually only attends to a few visual tokens around the target keypoint. And some keypoint tokens (such as the eye and the nose) may attend to close-by or even the same visual tokens. Thus, it is difficult to treat the attention values of each keypoint separately. For simplicity, as all human keypoints make up a rough human body area, we use $\mathbf{a} = \sum_j \mathbf{a}^j$ as the criterion to select visual tokens, which is the summation of all joints' attention maps. In detail, we keep visual tokens with the K largest corresponding values in $\mathbf{a}$ as the human tokens, and prune the remaining tokens. As a result, only K visual tokens and J keypoint tokens are sent to the following layers.

3.2 Multi-view Pose Estimation with PPT

Human Area Fusion. We propose the concept of *Human area fusion* for cross-view fusion in multi-view pose estimation, which considers pixels where human appears as corresponding candidates. Suppose there are m cameras, and each view maintains n pixels (tokens) in its feature map. We summarize three typical types of cross-view fusion strategies in Fig.1. 1) For global fusion, each pixel in each view calculates attention with respect to all n pixels in feature maps of other $m - 1$ views. Thus the computational complexity is $\mathcal{O}(m^2 n^2)$. 2) For epipolar-based fusion, each pixel in each view calculates attention with $k(k \ll n)$ pixels along the corresponded epipolar lines of other $m - 1$ views. Thus the computational complexity is $\mathcal{O}(m^2 nk)$. 3) For our human area fusion, we firstly select k' human foreground pixels in each view. Then we perform dense attention among these foreground tokens. As we also reduce the number of query pixels, the computational complexity is $\mathcal{O}(m^2 k'^2)$. Typically, $k < k' \ll n$. Thus, our method is an efficient way to perform cross-view fusion. Moreover, it also avoids the useless or even disturbing information from the background tokens and thus makes the model focus on the constraints within the human body.

Multi-view PPT.   Naturally, we can apply an off-the-shelf segmentation network [18] to obtain human foreground pixels and then perform human area fusion. However, a large amount of densely annotated images are required to train a segmentation model. To this end, we utilize PPT to efficiently locate a rough human foreground area without any mask labels, and further propose the *multi-view PPT* for multi-view pose estimation. Specifically, we design our network in a two-stage paradigm, as shown in Fig.4. Given the image $\mathbf{I}^m$ in each view, the share-weight PPT firstly produces selected human tokens $\tilde{\mathbf{X}}_v^m$ and keypoint tokens $\mathbf{X}_k^m$. Then we concatenate tokens from all views together and perform the dense attention among them with B transformer encoder layers. To help the network perceive the 3D space information, we also add the 3D positional encodings [36] on all selected visual tokens. Thus, each keypoint token can fuse visual information from all views. Moreover, it can learn correspondence constraints between keypoints both in the same view and among different views. Finally, a share-weight MLP head is placed on top of the keypoint token of each view to predicts keypoint heatmaps.

4   Experiments on monocular image

4.1   Settings

Datasets & Evaluation Metrics.   We firstly evaluate PPT on monocular 2D human pose estimation benchmarks. COCO [34] contains $200K$ images in the wild and $250K$ human instances with 17 keypoints. Following top-down methods [63, 50, 31], we crop human instances with the ground truth bounding boxes for training and with the bounding boxes provided by SimpleBaseline [63] for inference. The evaluation is based on object keypoint similarity, which measures the distance between the detected keypoint and the corresponding ground truth. The standard average precision (AP) and recall (AR) scores are reported. MPII [1] contains about $25K$ images and $40K$ human instances with 16 keypoints. The evaluation is based on the head-normalized probability of correct keypoint (PCKh) score [1]. A keypoint is correct if it falls within a predefined threshold to the groundtruth location. We report the PCKh@0.5 score by convention.

Implementation Details.   For fair comparison, we build our PPT based upon TokenPose-S, TokenPose-B, and TokenPose-L/D6 [31], namely PPT-S, PPT-B, and PPT-L/D6, respectively. For PPT-S and PPT-B, the number of encoder layers L is set to 12, the embedding size D is set to 192, the number of heads H is set to 8. They take the shallow stem-net and the HRNet-W32 as the CNN backbone, respectively. Following [44, 32], the HTI is performed $e = 3$ times and is inserted before the 4th, 7th, and 10th encoder layers. The PPT-L/D6 has $L = 12$ encoder layers and takes HRNet-W48 as the backbone. the HTI is inserted before the 2th, 4th, and 5th encoder layers. The number of visual tokens N_v is 256 for all networks, and the keep ratio r is set to 0.7 by default. Thus, only 88 visual tokens are left after three rounds pruning. We follow the same training

Method	#Params	GFLOPs	GFLOPs^T	AP	AP^{50}	AP^{75}	AP^M	AP^L	AR
SimpleBaseline-Res50 [63]	34M	8.9	-	70.4	88.6	78.3	67.1	77.2	76.3
SimpleBaseline-Res101 [63]	53M	12.4	-	71.4	89.3	79.3	68.1	78.1	77.1
SimpleBaseline-Res152 [63]	68.6M	15.7	-	72.0	89.3	79.8	68.7	78.9	77.8
HRNet-W32 [50]	28.5M	7.1	-	74.4	90.5	81.9	70.8	81.0	79.8
HRNet-W48 [50]	63.6M	14.6	-	75.1	90.6	82.2	71.5	81.8	80.4
Lite-HRNet-18 [69]	1.1M	0.20	-	64.8	86.7	73.0	62.1	70.5	71.2
Lite-HRNet-30 [69]	1.8M	0.31	-	67.2	88.0	75.0	64.3	73.1	73.3
EfficientPose-B [72]	3.3M	1.1	-	71.1	-	-	-	-	-
EfficientPose-C [72]	5.0M	1.6	-	71.3	-	-	-	-	-
TransPose-R-A4 [67]	6.0M	8.9	3.38	72.6	89.1	79.9	68.8	79.8	78.0
TransPose-H-S [67]	8.0M	10.2	4.88	74.2	89.6	80.8	70.6	81.0	79.5
TransPose-H-A6 [67]	17.5M	21.8	11.4	75.8	90.1	82.1	71.9	82.8	80.8
TokenPose-S [31]	6.6M	2.2	1.44	72.5	89.3	79.7	68.8	79.6	78.0
TokenPose-B [31]	13.5M	5.7	1.44	74.7	89.8	81.4	71.3	81.4	80.0
TokenPose-L/D6 [31]	20.8M	9.1	0.72	75.4	90.0	81.8	71.8	82.4	80.4
PPT-S (ours)	6.6M	1.6(**-27%**)	0.89(-38%)	72.2(**-0.3**)	89.0	79.7	68.6	79.3	77.8
PPT-B (ours)	13.5M	5.0(**-12%**)	0.89(-38%)	74.4(**-0.3**)	89.6	80.9	70.8	81.4	79.6
PPT-L/D6 (ours)	20.8M	8.7(**-4%**)	0.50(-31%)	75.2(**-0.2**)	89.8	81.7	71.7	82.1	80.4

Table 1. Results on COCO validation dataset. The input size is 256×192. GFLOPs^T means the GFLOPs for the transformers only following equations from [29], as our method only focus on accelerating the transformers.

Method	#Params	GFLOPs	Head	Sho	Elb	Wri	Hip	Kne	Ank	Mean
SimpleBaseline-Res50 [63]	34M	12.0	96.4	95.3	89.0	83.2	88.4	84.0	79.6	88.5
SimpleBaseline-Res101 [63]	53M	16.5	96.9	95.9	89.5	84.4	88.4	84.5	80.7	89.1
SimpleBaseline-Res152 [63]	53M	21.0	97.0	95.9	90.0	85.0	89.2	85.3	81.3	89.6
HRNet-W32. [50]	28.5M	9.5	96.9	96.0	90.6	85.8	88.7	86.6	82.6	90.1
TokenPose-S [31]	7.7M	2.5	96.0	94.5	86.5	79.7	86.7	80.1	75.2	86.2
PPT-S	7.7M	1.9 (**-24%**)	96.6	94.9	87.6	81.3	87.1	82.4	76.7	87.3 (+**1.1**)
TokenPose-B [31]	14.4M	7.1	97.0	96.1	90.1	85.6	89.2	86.1	80.3	89.7
PPT-B	14.4M	6.2 (**-13%**)	97.0	95.7	90.1	85.7	89.4	85.8	81.2	89.8 (+**0.1**)

Table 2. Results on the MPII validation set (PCKh@0.5). The input size is 256×256.

recipes as [31]. In detail, all networks are optimized by Adam optimizer [25] with Mean Square Error (MSE) loss for 300 epochs. The learning rate is initialized with 0.001 and decays at the 200-th and the 260-th epoch with ratio 0.1. As locating human is difficult at early training stages, the keep ratio is gradually reduced from 1 to r with a cosine schedule during the early 100 epochs.

4.2   Results

The results are shown in Table 1 and Table 2 for COCO and MPII, respectively. Generally, the transformer-based methods [31, 67] maintain less number of parameters. On COCO, compared with the TokenPose, PPT achieves significant acceleration while matching its accuracy. For example, PPT-S reduces 27% total inference FLOPs while only reducing 0.3 AP. Compared to SimpleBaseline-ResNet152 [63], PPT-S achieves equal performance but only requires 10% FLOPS. We can also observe consistent conclusion on PPT-B and

PPT-L. Note that, for PPT-B and PPT-L, the CNN backbone takes a large portion of computation. Thus, the reduction of total FLOPs is relatively small. Meanwhile, compared with other efficient pose estimation networks [69, 72], the AP of PPT-S is 72.2, which is much better than EfficientPose-C [72] with 71.3 AP at the same FLOPs level. More over, On MPII, our PPT-S can even improve on the PCKh of TokenPose-S by 1.1%. We believe that slimming the number of tokens can also make the attention focus on key elements [76]. Thus, our PPT is efficient yet powerful, and it is applicable to any TokenPose variants. All of these results suggest that pruning background tokens does not hurt the overall accuracy and calculating attention among human foreground tokens is sufficient for 2D human pose estimation.

4.3   Visualizations

We visualize the selected tokens from PPT-S in Fig. 5. We present the original images and the selected tokens at different layers. Remarkably, the human areas are gradually refined as the network deepens. The final selected tokens can be considered as a rough human mask. Thus, our HTI can successfully locate human tokens as expected. Moreover, the HTI can handle quite a few complicated situations such as man-object interaction (Fig.5(b)), oblique body pose (Fig. 5(c)), occlusion (Fig. 5(d)), and multiple persons (Fig.5(e) 5(f)). Nevertheless, when only part of human body appears in the image (Fig.5(g)5(h)), the quality of the located human mask could be imperfect. In these cases, we hypothesize that some keypoint tokens such as ankle and knee cannot locate the corresponding joints as they are invisible. Thus, they may just give equal attention score, which leads to inaccurate token selection.

4.4   Ablation Studies

The keep ratio r controls the trade-off between the acceleration and the accuracy. Meanwhile, reducing tokens also introduces some regularization [76]. We take PPT-S and vary r from 0.6 to 0.8 on both COCO and MPII. The results are shown in Table 3. The reduction of AP is always less than 1%. When the r is relatively small, PPT can achieve considerable speedup but may not cover the entire human body. As a result, the accuracy of pose estimation is slightly dropped. To maintain the accuracy, we choose 0.7 as our default keep ratio.

Method	Keep Ratio	# Visual Tokens	COCO			MPII		
			AP	AR	FLOPs	PCKh@0.5	PCKh@0.1	FLOPs
TokenPose-S	1.0	256 (100%)	72.5	78.0	2.23	86.2	32.2	2.53
PPT-S	0.8	131 (51%)	72.0 (-0.5)	77.6 (-0.4)	1.75 (-22%)	86.9 (+0.7)	32.9 (+0.7)	2.06 (-19%)
PPT-S	0.7	88 (34%)	72.2 (-0.3)	77.8 (-0.2)	1.61 (-27%)	87.3 (+1.1)	34.1 (+1.9)	1.92 (-24%)
PPT-S	0.6	56 (22%)	71.8 (-0.7)	77.5 (-0.5)	1.52 (-32%)	86.7 (+0.5)	32.3 (+0.1)	1.82 (-28%)

Table 3. Results of PPT-S on COCO and MPII with different keep ratio r.

Fig. 5. Visualizations of the selected tokens at each HTI module on COCO. The masked regions represent the pruned tokens (We use blue circles to mask out face for privacy issue). For each image group, the first column is the original image, the 2nd, 3rd, and 4th colums are the selected tokens by HTI at the 4th, 7th, and 10th layers, respectively.

5 Experiments on Multi-view Pose Estimation

5.1 Settings

Datasets & Evaluation Metrics. We evaluate multi-view PPT on two single-person datasets of multi-view 3D human pose estimation, *i.e.*, Human 3.6M [22, 4] and Ski-Pose [49, 16] [4]. Human 3.6M contains video frames captured by $M = 4$ indoor cameras. It includes many daily activities such as eating and discussion. We follow the same train-test split as in [43, 23, 19], where subjects $1, 5, 6, 7, 8$ are used for training, and $9, 11$ are for testing. We also exclude some scenes of $S9$ from the evaluation as their 3D annotations are damaged [23]. Ski-Pose contains video frames captured by outdoor cameras. It is created to help analyze skiers's giant slalom. There are $8,481$ and $1,716$ frames in the training and testing sets,

---

[4] Only authors from UCI downloaded and accessed these two datasets. Authors from Tencent and Meta don't have access to them.

Method	#V	MACs	shlder	elb	wri	hip	knee	ankle	root	belly	neck	nose	head	Avg
ResNet50 [63]	1	51.7G	97.0	91.9	87.3	99.4	95.0	90.8	100.0	98.3	99.4	99.3	99.5	95.2
TransPose [67]	1	43.6G	96.0	92.9	88.4	99.0	95.0	91.8	100.0	97.5	99.0	99.4	99.6	95.3
TokenPose [31]	1	11.2G	96.0	91.3	85.8	99.4	95.2	91.5	100.0	98.1	99.1	99.4	99.1	94.9
Epipolar Transformer [19]	2	51.7G	97.0	93.1	91.8	99.1	96.5	91.9	100.0	99.3	99.8	99.8	99.3	96.3
TransFusion [36]	2	50.2G	97.2	96.6	93.7	99.0	96.8	91.7	100.0	96.5	98.9	99.3	99.5	96.7
Crossview Fusion [43]	4	55.1G	97.2	94.4	92.7	**99.8**	97.0	92.3	100.0	98.5	99.1	99.1	99.1	96.6
TokenPose+Transformers	4	11.5G	97.1	97.3	95.2	99.2	98.1	93.1	100.0	98.8	99.2	99.3	99.1	97.4
PPT	1	9.6G	96.0	91.8	86.5	99.2	95.6	92.2	100.0	98.4	99.3	99.5	99.4	95.3
Multi-view PPT	2	9.7G	97.1	95.5	91.9	99.4	96.4	92.1	100.0	99.0	99.2	99.3	99.0	96.6
Multi-view PPT	4	9.7G	97.6	**98.0**	96.4	99.7	98.4	93.8	100.0	99.0	**99.4**	**99.5**	**99.5**	97.9
Multi-view PPT + 3DPE	4	9.7G	**98.0**	**98.0**	96.4	99.7	**98.5**	**94.0**	100.0	99.1	99.2	99.4	99.3	**98.0**

Table 4. 2D pose estimation on Human3.6M. The metric is JDR on original image. All inputs are resized to 256×256. #V means the number of views used in cross-view fusion step. The FLOPs is the total computation for each view and cros-view fusion.

Method	Dir	Disc	Eat	Greet	Phone	Pose	Purch	Sit	SitD	Smoke	Photo	Wait	WalkD	Walk	WalkT	Avg
Crossview Fusion[43]	24.0	28.8	25.6	24.5	28.3	24.4	26.9	30.7	34.4	29.0	32.6	25.1	24.3	30.8	24.9	27.8
Epipolar Trans. [19]	23.2	27.1	23.4	22.4	32.4	**21.4**	**22.6**	37.3	35.4	29.0	27.7	24.2	21.2	26.6	**22.3**	27.1
TransFusion [36]	24.4	**26.4**	23.4	**21.1**	25.2	23.2	24.7	33.8	**29.8**	26.4	26.8	24.2	23.2	26.1	23.3	25.8
Multi-PPT+3DPE	**21.8**	26.5	**21.0**	22.4	**23.7**	23.1	23.2	**27.9**	30.7	**24.6**	**26.7**	**23.3**	**21.2**	**25.3**	22.6	**24.4**

Table 5. The MPJPE of each pose sequence on Human 3.6M.

respectively. We use the Joint Detection Rate (JDR) on original images [43] to evaluate the 2D pose accuracy. JDR measures the percentage of successfully detected keypoints within a predefined distance of the ground truth location. The 3D pose is evaluated by Mean Per Joint Position Error (MPJPE) between the ground truth 3D pose in world coordinates and the estimated 3D pose.

Implementation Details. We build multi-view PPT upon PPT-S. The first 9 transformer layers are used to extract human tokens, and the last 3 transformer layers are used for cross-view fusion. Thus, no additional parameters are introduced. Following the settings in [19, 36], we start from a PPT-S pre-trained on COCO and finetune it on multi-view human pose datasets, as it is difficult to train the transformer from scratch with examples in limited scenes. We apply Adam optimizer and train the model for 20 epochs with MSE loss. The learning rate starts with 0.001 and later on decays at 10-th and 15-th epoch with ratio 0.1. The keep ratio r is set to 0.7 through the entire training process. We resize input images to 256×256 and follow the same data augmentation in [43, 36].

5.2   Results

The 2D results on Human 3.6m is shown in Table 4. The MACs (multiply-add operations) consider both single-view forward MACs of all views and cross-view fusion MACs. Noticeably, our multi-view PPT outperforms all previous cross-view fusion methods on JDR. The JDR can be further improved with the 3D

Method	MACs	2D Pose / JDR (%) ↑	3D Pose / MPJPE (mm) ↓
Simple Baseline-Res50 [63]	77.6G	94.5	39.6
TokenPose [31]	16.8G	95.0	35.6
Epipolar Transformer [19]	77.6G	94.9	34.2
Multi-view PPT	**14.5G**	**96.3**	**34.1**

Table 6. 2D and 3D pose estimation accuracy comparison on Ski-Pose.

Fig. 6. Visualizations of the final located tokens on Human 3.6M validation set. For each group, each column is an image from one view. The masked regions represent the pruned tokens. We perform cross-view fusion among these selected tokens.

positional encodings (3DPE) [36] on visual tokens. Meanwhile, it can significantly reduce the computation of all 4 view fusion, *i.e.*, the MACs is reduced from 55.1G to 9.7G. When only fusing 2 views, multi-view PPT still achieves comparable accuracy with other two-view-fusion methods [19, 36], Moreover, we add the baseline that adds transformers on top of TokenPose to perform cross-view fusion, which can be considered as multi-view PPT without token pruning. The JDR is 97.4% (-0.7% with respect to our multi-view PPT), which supports that our human area fusion is better than global attention in both accuracy and efficiency. The MPJPE of estimated 3D pose is reported in Table 5. We can observe that multi-view PPT also achieves the best MPJPE on 3D pose, especially on sophisticated action sequences such as "Phone" and "Smoke", as the result of 3D pose is determined by the accuracy of 2D pose. Therefore, our "human area fusion" strategy is better than previous fusion strategies as it strikes a good balance between efficiency and accuracy. We can also observe consistent conclusion on Ski-Pose from Table 6. Nevertheless, it seems that the performance in this datatset tends to be saturated. The reason might be that there is limited number of training examples, thus the transformer is easy to overfit.

5.3   Visualizations

Human Tokens. Fig.6 presents the selected human tokens in all views. Similar to the conclusion on COCO, our PPT accurately locates all human areas and prunes background areas in all views. Moreover, the tokens used in the cross-view fusion step can be significantly reduced.

Qualitative results. We present examples of predicted 2D heatmaps on the image in Fig.7, and compare our methods with TransFusion [36]. It is observed that our method can solve heavy occlusion cases very well, while TransFusion cannot. For two-view-fusion method, occlusion cases in current view may still be
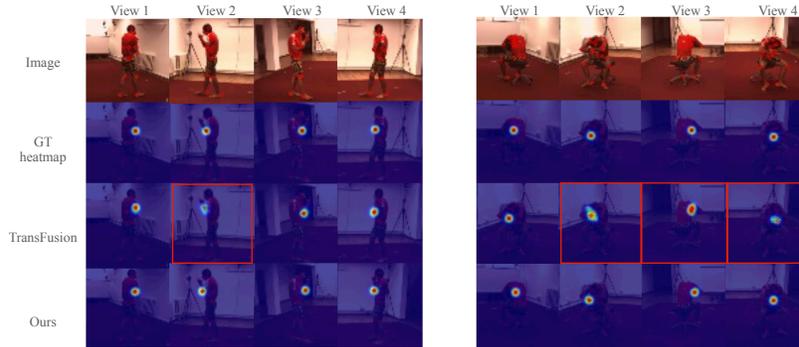
Fig. 7. Sample heatmaps of our approach.
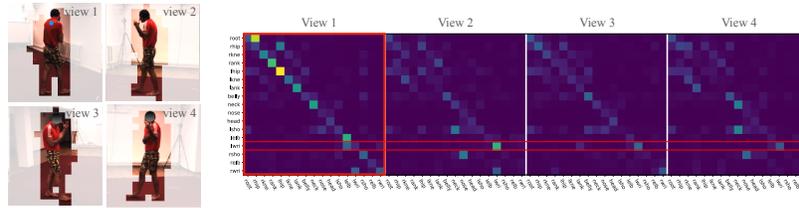
Fig. 8. Attention maps among keypoint tokens.

occluded in the neighbor view. For example, the heatmap marked with red box is inaccurate in both view 2 and view 4. Thus, fusing this bad quality heatmap cannot improve the final prediction. However, our method can avoid this problem by fusing clues from all views.

Attentions.   We present an example of the attention map between keypoint tokens in Fig.8. Given keypoint tokens in one view, they pay attention to keypoints tokens in all views. For example, the left wrist in the first view (blue dot) is occluded, thus its corresponded keypoint token attends to the keypoint token in the second view, where the keypoint is visible. Therefore, the keypoint token in multi-view PPT can learn the dependencies among joints in different views.

6   Conclusion

In this paper, we propose the PPT for 2D human pose estimation. Experiments on COCO and MPII show that the PPT achieves similar accuracy compared with previous transformer-based networks but reduces the computation significantly. We also empirically show that PPT can locate a rough human mask as expected. Furthermore, we propose the multi-view PPT to perform the cross-view fusion among human areas. We demonstrate that multi-view PPT efficiently fuses cues from many views and outperforms previous cross-view fusion methods on Human 3.6M and Ski-Pose.

References

1. Andriluka, M., Pishchulin, L., Gehler, P., Schiele, B.: 2d human pose estimation: New benchmark and state of the art analysis. In: CVPR (2014)
2. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: ECCV (2020)
3. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: ICCV (2021)
4. Catalin Ionescu, Fuxin Li, C.S.: Latent structured models for human pose estimation. In: ICCV (2011)
5. Chen, C.F., Fan, Q., Panda, R.: Crossvit: Cross-attention multi-scale vision transformer for image classification. ICCV (2021)
6. Chen, L., Lin, S.Y., Xie, Y., Lin, Y.Y., Xie, X.: Mvhm: A large-scale multi-view hand mesh benchmark for accurate 3d hand pose estimation. In: WACV. pp. 836–845 (2021)
7. Chen, T., Cheng, Y., Gan, Z., Yuan, L., Zhang, L., Wang, Z.: Chasing sparsity in vision transformers: An end-to-end exploration. NeurIPS (2021)
8. Chen, T., Zhang, Z., Cheng, Y., Awadallah, A., Wang, Z.: The principle of diversity: Training stronger vision transformers calls for reducing all levels of redundancy. In: CVPR (2022)
9. Chen, X., Cao, Q., Zhong, Y., Zhang, J., Gao, S., Tao, D.: Dearkd: Data-efficient early knowledge distillation for vision transformers. In: CVPR (2022)
10. Chen, Y., Ma, H., Kong, D., Yan, X., Wu, J., Fan, W., Xie, X.: Nonparametric structure regularization machine for 2d hand pose estimation. In: WACV (2020)
11. Chen, Y., Ma, H., Wang, J., Wu, J., Wu, X., Xie, X.: Pd-net: Quantitative motor function evaluation for parkinson's disease via automated hand gesture analysis. In: KDD (2021)
12. Chen, Y., Wang, Z., Peng, Y., Zhang, Z., Yu, G., Sun, J.: Cascaded pyramid network for multi-person pose estimation. In: CVPR (2018)
13. Chu, X., Yang, W., Ouyang, W., Ma, C., Yuille, A.L., Wang, X.: Multi-context attention for human pose estimation. In: CVPR (2017)
14. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. ICLR (2021)
15. Fang, Y., Liao, B., Wang, X., Fang, J., Qi, J., Wu, R., Niu, J., Liu, W.: You only look at one sequence: Rethinking transformer in vision through object detection. NeurIPS (2021)
16. Fasel, B., Spörri, J., Chardonnens, J., Kröll, J., Müller, E., Aminian, K.: Joint inertial sensor orientation drift reduction for highly dynamic movements. IEEE journal of biomedical and health informatics (2017)
17. Han, S., Mao, H., Dally, W.J.: Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. ICLR (2016)
18. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: ICCV (2017)
19. He, Y., Yan, R., Fragkiadaki, K., Yu, S.I.: Epipolar transformers. In: CVPR (2020)
20. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531 (2015)
21. Huang, Z., Wan, C., Probst, T., Van Gool, L.: Deep learning on lie groups for skeleton-based action recognition. In: CVPR (2017)
22. Ionescu, C., Papava, D., Olaru, V., Sminchisescu, C.: Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. IEEE Transactions on Pattern Analysis and Machine Intelligence (2014)

23. Iskakov, K., Burkov, E., Lempitsky, V., Malkov, Y.: Learnable triangulation of human pose. In: ICCV (2019)
24. Ke, L., Chang, M.C., Qi, H., Lyu, S.: Multi-scale structure-aware network for human pose estimation. In: ECCV (2018)
25. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. ICLR (2015)
26. Kong, D., Chen, Y., Ma, H., Yan, X., Xie, X.: Adaptive graphical model network for 2d handpose estimation. BMVC (2019)
27. Kong, D., Ma, H., Chen, Y., Xie, X.: Rotation-invariant mixed graphical model network for 2d hand pose estimation. In: WACV (2020)
28. Kong, D., Ma, H., Xie, X.: Sia-gcn: A spatial information aware graph neural network with 2d convolutions for hand pose estimation. BMVC (2020)
29. Kong, Z., Dong, P., Ma, X., Meng, X., Niu, W., Sun, M., Ren, B., Qin, M., Tang, H., Wang, Y.: Spvit: Enabling faster vision transformers via soft token pruning. arXiv preprint arXiv:2112.13890 (2021)
30. Li, K., Wang, S., Zhang, X., Xu, Y., Xu, W., Tu, Z.: Pose recognition with cascade transformers. CVPR (2021)
31. Li, Y., Zhang, S., Wang, Z., Yang, S., Yang, W., Xia, S.T., Zhou, E.: Tokenpose: Learning keypoint tokens for human pose estimation. In: ICCV (2021)
32. Liang, Y., GE, C., Tong, Z., Song, Y., Wang, J., Xie, P.: EVit: Expediting vision transformers via token reorganizations. In: ICLR (2022)
33. Lin, K., Wang, L., Liu, Z.: End-to-end human pose and mesh reconstruction with transformers. CVPR (2021)
34. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV (2014)
35. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: ICCV (2021)
36. Ma, H., Chen, L., Kong, D., Wang, Z., Liu, X., Tang, H., Yan, X., Xie, Y., Lin, S.Y., Xie, X.: Transfusion: Cross-view fusion with transformer for 3d human pose estimation. BMVC (2021)
37. Mao, W., Ge, Y., Shen, C., Tian, Z., Wang, X., Wang, Z.: Tfpose: Direct human pose estimation with transformers. arXiv preprint arXiv:2103.15320 (2021)
38. Meng, L., Li, H., Chen, B.C., Lan, S., Wu, Z., Jiang, Y.G., Lim, S.N.: Adavit: Adaptive vision transformers for efficient image recognition. In: CVPR (2022)
39. Neff, C., Sheth, A., Furgurson, S., Tabkhi, H.: Efficienthrnet: Efficient scaling for lightweight high-resolution multi-person pose estimation. arXiv preprint arXiv:2007.08090 (2020)
40. Newell, A., Yang, K., Deng, J.: Stacked hourglass networks for human pose estimation. In: ECCV (2016)
41. Osokin, D.: Real-time 2d multi-person pose estimation on cpu: Lightweight openpose. arXiv preprint arXiv:1811.12004 (2018)
42. Papandreou, G., Zhu, T., Chen, L.C., Gidaris, S., Tompson, J., Murphy, K.: Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model. In: ECCV (2018)
43. Qiu, H., Wang, C., Wang, J., Wang, N., Zeng, W.: Cross view fusion for 3d human pose estimation. In: ICCV (2019)
44. Rao, Y., Zhao, W., Liu, B., Lu, J., Zhou, J., Hsieh, C.J.: Dynamicvit: Efficient vision transformers with dynamic token sparsification. NeurIPS (2021)
45. Ryoo, M., Piergiovanni, A., Arnab, A., Dehghani, M., Angelova, A.: Tokenlearner: Adaptive space-time tokenization for videos. NeurIPS (2021)
46. Shen, S., Dong, Z., Ye, J., Ma, L., Yao, Z., Gholami, A., Mahoney, M.W., Keutzer, K.: Q-bert: Hessian based ultra low precision quantization of bert. In: AAAI (2020)

47. Shen, X., Yuan, G., Niu, W., Ma, X., Guan, J., Li, Z., Ren, B., Wang, Y.: Towards fast and accurate multi-person pose estimation on mobile devices. arXiv preprint arXiv:2106.15304 (2021)
48. Simon, T., Joo, H., Matthews, I., Sheikh, Y.: Hand keypoint detection in single images using multiview bootstrapping. In: CVPR (2017)
49. Spörri, J.: Reasearch dedicated to sports injury prevention-the'sequence of prevention'on the example of alpine ski racing. Habilitation with Venia Docendi in Biomechanics (2016)
50. Sun, K., Xiao, B., Liu, D., Wang, J.: Deep high-resolution representation learning for human pose estimation. In: CVPR (2019)
51. Sun, M., Ma, H., Kang, G., Jiang, Y., Chen, T., Ma, X., Wang, Z., Wang, Y.: Vaqf: Fully automatic software-hardware co-design framework for low-bit vision transformer. arXiv preprint arXiv:2201.06618 (2022)
52. Tompson, J.J., Jain, A., LeCun, Y., Bregler, C.: Joint training of a convolutional network and a graphical model for human pose estimation. NIPS (2014)
53. Toshev, A., Szegedy, C.: Deeppose: Human pose estimation via deep neural networks. In: CVPR (2014)
54. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. ICML (2021)
55. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. NIPS (2017)
56. Wang, X., Girshick, R., Gupta, A., He, K.: Non-local neural networks. In: CVPR (2018)
57. Wang, Y., Li, M., Cai, H., Chen, W.M., Han, S.: Lite pose: Efficient architecture design for 2d human pose estimation. In: CVPR (2022)
58. Wang, Y., Xu, Z., Wang, X., Shen, C., Cheng, B., Shen, H., Xia, H.: End-to-end video instance segmentation with transformers. CVPR (2021)
59. Wang, Z., Yang, J., Fowlkes, C.: The best of both worlds: Combining model-based and nonparametric approaches for 3d human body estimation. In: CVPR ABAW workshop (2022)
60. Wang, Z., Chen, L., Rathore, S., Shin, D., Fowlkes, C.: Geometric pose affordance: 3d human pose with scene constraints. arXiv preprint arXiv:1905.07718 (2019)
61. Wang, Z., Shin, D., Fowlkes, C.C.: Predicting camera viewpoint improves cross-dataset generalization for 3d human pose estimation. In: ECCV 3DPW workshop (2020)
62. Wei, S.E., Ramakrishna, V., Kanade, T., Sheikh, Y.: Convolutional pose machines. In: CVPR (2016)
63. Xiao, B., Wu, H., Wei, Y.: Simple baselines for human pose estimation and tracking. In: ECCV (2018)
64. Xie, R., Wang, C., Wang, Y.: Metafuse: A pre-trained fusion model for human pose estimation. In: CVPR (2020)
65. Yan, S., Xiong, Y., Lin, D.: Spatial temporal graph convolutional networks for skeleton-based action recognition. In: AAAI (2018)
66. Yan, X., Tang, H., Sun, S., Ma, H., Kong, D., Xie, X.: After-unet: Axial fusion transformer unet for medical image segmentation. In: WACV (2022)
67. Yang, S., Quan, Z., Nie, M., Yang, W.: Transpose: Keypoint localization via transformer. In: ICCV (2021)
68. You, C., Zhao, R., Liu, F., Chinchali, S., Topcu, U., Staib, L., Duncan, J.S.: Class-aware generative adversarial transformers for medical image segmentation. arXiv preprint arXiv:2201.10737 (2022)

69. Yu, C., Xiao, B., Gao, C., Yuan, L., Zhang, L., Sang, N., Wang, J.: Lite-hrnet: A lightweight high-resolution network. In: CVPR (2021)
70. Yu, S., Chen, T., Shen, J., Yuan, H., Tan, J., Yang, S., Liu, J., Wang, Z.: Unified visual transformer compression. In: ICLR (2022)
71. Yuan, L., Chen, Y., Wang, T., Yu, W., Shi, Y., Jiang, Z.H., Tay, F.E., Feng, J., Yan, S.: Tokens-to-token vit: Training vision transformers from scratch on imagenet. In: ICCV (2021)
72. Zhang, W., Fang, J., Wang, X., Liu, W.: Efficientpose: Efficient human pose estimation with neural architecture search. Computational Visual Media (2021)
73. Zhang, Z., Wang, C., Qiu, W., Qin, W., Zeng, W.: Adafuse: Adaptive multiview fusion for accurate human pose estimation in the wild. IJCV (2021)
74. Zheng, C., Zhu, S., Mendieta, M., Yang, T., Chen, C., Ding, Z.: 3d human pose estimation with spatial and temporal transformers. ICCV (2021)
75. Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., Fu, Y., Feng, J., Xiang, T., Torr, P.H., et al.: Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. CVPR (2021)
76. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable detr: Deformable transformers for end-to-end object detection. ICLR (2021)