Human Pose as Compositional Tokens

Zigang Geng[1,3] , Chunyu Wang[3*], Yixuan Wei[2,3], Ze Liu[1,3], Houqiang Li[1], Han Hu[3*]
[1]University of Science and Technology of China [2]Tsinghua University [3]Microsoft Research Asia
https://sites.google.com/view/pctpose

Abstract

Human pose is typically represented by a coordinate vector of body joints or their heatmap embeddings. While easy for data processing, unrealistic pose estimates are admitted due to the lack of dependency modeling between the body joints. In this paper, we present a structured representation, named Pose as Compositional Tokens (PCT), to explore the joint dependency. It represents a pose by M discrete tokens with each characterizing a sub-structure with several interdependent joints (see Figure 1). The compositional design enables it to achieve a small reconstruction error at a low cost. Then we cast pose estimation as a classification task. In particular, we learn a classifier to predict the categories of the M tokens from an image. A pre-learned decoder network is used to recover the pose from the tokens without further post-processing. We show that it achieves better or comparable pose estimation results as the existing methods in general scenarios, yet continues to work well when occlusion occurs, which is ubiquitous in practice. The code and models are publicly available at https://github.com/Gengzigang/PCT.

1. Introduction

Human pose estimation is a fundamental task in computer vision which aims to estimate the positions of body joints from images. The recent progress has focused on network structures [74, 87, 96], training methods [31, 68, 93], and fusion strategies [14, 15, 61, 67, 84, 102], which have notably advanced the accuracy on public datasets. However, it remains an open problem in challenging scenarios, *e.g.*, in the presence of occlusion, which hinders its application in practice.

Current 2/3D pose estimators usually represent a pose by a coordinate vector [23, 34, 79, 110] or its heatmap embeddings [40, 55, 60, 74, 75, 80, 87, 90]. In both representations, the joints are treated independently, ignoring the fact that the body joints can serve as mutual context to each other. As a result, they may get unrealistic estimates when occlusion occurs as shown in Figure 2 (top). However, it is interesting to note that humans can easily predict intact poses from only the visible joints and the visual features. This is probably because people are able to use context to aid recognition as evidenced by some psychology experiments [5, 58]. Some works attempt to introduce a tree or graph structure [2, 21, 65, 85] to model joint dependency. However, the hand-designed rules usually make unrealistic assumptions on the relationships, making them incapable to represent complex patterns.

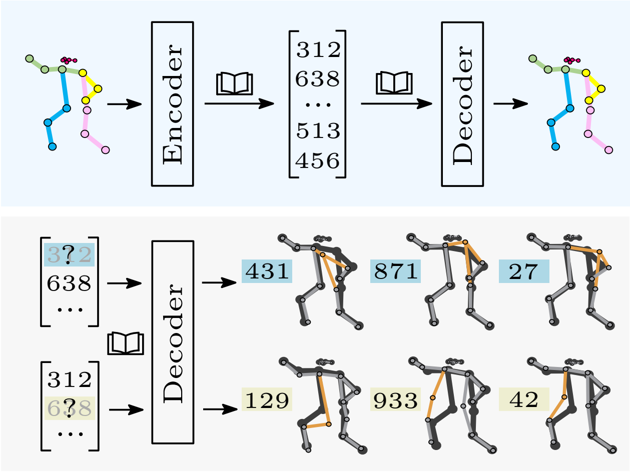

Figure 1. Our approach represents a pose by M discrete tokens which are indices to the codebook entries (**top**). Each token is learned to represent a sub-structure. In each row, we show that if we change the state of one token to different values, it consistently changes the same sub-structure highlighted by orange. The black poses are before changing (**bottom**).

In this work, we hope to learn the dependency between the joints earlier in the representation stage without any assumptions. Our initial idea is to learn a set of prototype poses that are realistic, and represent every pose by the nearest prototype. While it can guarantee that all poses are realistic, it requires a large number of prototypes to reduce the quantization error to a reasonable level which is computa-

[*] Equal Advising

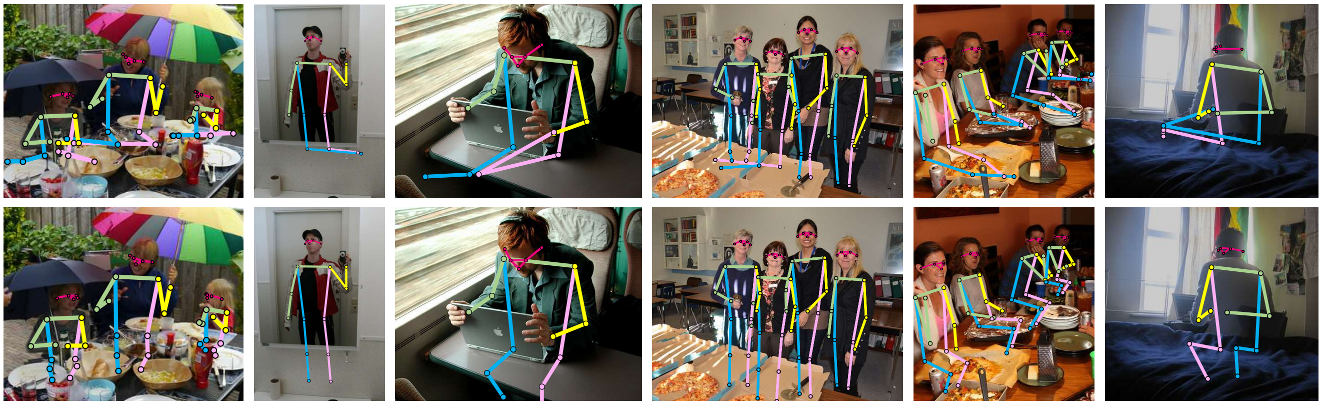

Figure 2. Heatmap-based method (**top**) v.s. our PCT method (**bottom**) in occluded scenes. PCT predicts reasonable poses even under severe occlusion. The images are from COCO val2017.

tionally infeasible. Instead, we propose a discrete representation, named pose as compositional tokens (PCT). Figure 3 shows the two stages of the representation. In Stage I, we learn a compositional encoder to transform a pose into M token features, with each encoding a sub-structure of the pose. See Figure 1 for some examples. Then the tokens are quantized by a shared codebook. So, a pose is simply represented by M discrete indices. The space represented by the codebook is sufficiently large to represent all poses accurately. We jointly learn the encoder, the codebook, and the decoder by minimizing a reconstruction error.

In Stage II, we cast human pose estimation as a classification task. Given an image, we predict the categories of the M tokens, from which the pose is recovered by the decoder network. The PCT representation has several advantages. First, the dependency between the joints is modeled by the tokens, which helps to reduce the chance of getting unrealistic pose estimates. In particular, we see evidence that it has the potential to obtain reasonable estimates even when a large portion of the body is occluded. See Figure 2 (bottom) for some examples. Second, it does not require any expensive post-processing modules such as UDP [29] which is required by the heatmap representation to reduce the quantization errors. Third, it provides a unified representation for 2D and 3D poses. In addition, the discrete representation potentially facilitates its interactions with other discrete modalities such as text and speech. But this is not the focus of this work.

We extensively evaluate our approach in 2D human pose estimation on five benchmark datasets. It gets better or comparable accuracy as the state-of-the-art methods on all of them. But more importantly, it achieves significantly better results when evaluated only on the occluded joints, validating the advantages of its dependency modeling capability. We also present the results in 3D pose estimation on the H36M dataset on which it achieves comparable accuracy as the state-of-the-art methods using a simple architecture. The results demonstrate that it has wide applicability.

2. Related works

In this section, we first briefly discuss the widely used pose representations. Then we discuss the methods that explore joint dependencies.

2.1. Pose representations

Coordinates. Early works [4, 9, 50, 56, 76, 79, 110] propose to directly regress the coordinates of body joints from images. While efficient, the accuracy is worse than the heatmap-based methods because it is challenging to learn the highly non-linear mapping. Some works [23, 89] propose to improve them by focusing on local features around the joints. Residual Log-likelihood Estimation [34] proposes a novel regression paradigm to capture the underlying output distribution. MDN [82] introduces mixture density network for regression. Recently, transformer [83] brings notable improvement [36, 49, 71] due to its ability to capture long-range information.

Heatmaps. The heatmap representation [3, 8, 20, 38, 41, 48, 54, 62, 69, 88, 92, 98] has been dominant since its introduction [6, 78, 90] because of its strong localization and generalization ability. Many follow-up works have been devoted to continuously improving them, including proposing powerful networks [7, 12, 13, 27, 55, 74] to estimate the heatmaps more accurately, introducing the attention operator to the models [40, 72, 97, 103], reducing the quantization errors [29, 105], fusion with the coordinate prediction-based methods [19, 25, 60, 75], refining the results [22, 53, 73, 85], leveraging other tasks [33, 57, 59], and leveraging large unlabeled datasets [32, 93]. However, the heatmap representation suffers from quantization errors caused by the downsampling operations in neural networks. Besides, the joint dependency is not modeled by the heatmaps.

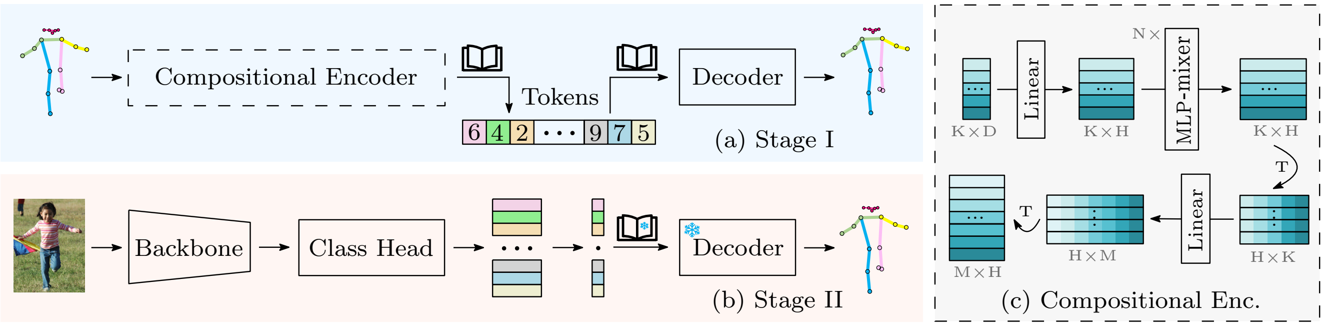

Figure 3. Two stages of the PCT representation (a,b) and the structure of the compositional encoder (c). In Stage I, we learn a compositional encoder to transform a pose into M tokens which are quantized by a codebook. So, a pose is represented by a set of discrete indices to the codebook. In Stage II, we cast pose estimation as a classification task by predicting the categories of the M tokens, *i.e*. the indices to the codebook entries. They will be decoded by a decoder network to obtain the final pose.

Discrete bins. Recent works [10, 39, 47] propose to divide each pixel into several bins, allowing sub-pixel localization accuracy. The horizontal and vertical coordinates of each joint are separately quantized into discrete classes. Similar to our work, they also cast human pose estimation as a classification task. However, each coordinate of the pose is treated independently which differs from our structured representation.

2.2. Modeling joint dependency

Since the human body has an articulated structure, there are many works trying to model joint dependency to help resolve low-level ambiguities. However, most of them focus on the modeling aspect rather than representation which is the focus of this work.

Pictorial structures. Some works [2, 21, 63, 65, 100] propose to use the deformable model where the relationship between body joints is explicitly considered based on anatomy priors (*e.g*. limb lengths). However, they have three disadvantages. First, they usually make strong assumptions on the relationships, *e.g*. Gaussian distribution on the offsets between two joints, making them incapable to represent complex patterns. Second, they still require that the body joints can be independently detected from images first, and based on that they use the dependency priors to obtain the most plausible configuration. However, the first step is already very difficult in cluttered scenes with serious occlusions. Finally, they cannot be trained end-to-end with the deep networks with an exception [78] that needs to relax the formulation.

Implicit modeling. The recent deep learning-based methods [16, 64, 85, 99, 101, 106] implicitly model the dependency by propagating the visual features between the joints. For example, Chu *et al*. [16] introduce geometrical transform kernels to fuse the features of different channels which are believed to characterize different joints. Wang *et al*. [85] use Graph Convolutional Network to refine pose estimates which are obtained by the heatmap-based methods first. In addition, Chen *et al*. [11] propose to learn a pose discriminator to exclude non-realistic pose estimates and push the predictor to learn poses with reasonable structures. Li *et al*. [40] explicitly learn a type embedding for each joint and apply the transformer to model the relationships among the joints. But from the aspect of representation, they still treat each joint independently and predict the heatmap for each joint.

Our PCT representation differs from the previous methods in three aspects. First, the joint dependency is encoded earlier in the representations by the tokens (changing the state of a token changes the corresponding sub-structure rather than a single joint). In contrast, the other three representations treat each joint independently. Second, the sub-structures are automatically learned from training data without making any unrealistic assumptions. We empirically show that it has a stronger capability to resolve ambiguities caused by occlusion in a variety of situations. Third, the joint dependency is explicitly imposed rather than by implicit feature propagation. The latter method still allows unrealistic pose estimates in challenging situations.

3. Pose as Compositional Tokens

In Section 3.1, we describe how to learn the codebook and the encoder/decoder networks. Section 3.2 explains how it is used in the human pose estimation task.

3.1. Learning compositional tokens

We represent a raw pose as $\mathbf{G} \in \mathbb{R}^{K \times D}$ where K is the number of body joints and D is the dimension of each joint, where $D = 2$ for 2D pose, and $D = 3$ for 3D pose, respectively. We learn a compositional encoder $f_e(\cdot)$ to transform a pose into M token features:

$$\mathbf{T} = (\mathbf{t}_1, \mathbf{t}_2, \cdots, \mathbf{t}_M) = f_e(\mathbf{G}), \tag{1}$$

where each token feature $\mathbf{t}_i \in \mathbb{R}^H$ approximately corresponds to a sub-structure of the pose which involves a few interdependent joints. Figure 1 shows some of the learned examples. Note that the representation has lots of redundancy because different tokens may have overlapping joints. The redundancy makes it robust to occlusions of individual parts.

Figure 3 (c) shows the network structure of the encoder. The position of each body joint is first fed to a linear projection layer to increase the feature dimension. Then the features are fed to a series of MLP-Mixer [77] blocks to deeply fuse the features of different joints. Finally, we extract M token features by applying a linear projection to the features across all of the joints.

Similar to [81], we define a latent embedding space by a codebook $\mathbf{C} = (\mathbf{c}_1, \cdots, \mathbf{c}_V)^{\mathrm{T}} \in \mathbb{R}^{V \times N}$ where V is the number of codebook entries. We quantize each token $\mathbf{t}_i$ by the nearest neighbor look-up using the embedding space as shown in the following equation:

$$q(\mathbf{t}_i = v|\mathbf{G}) = \begin{cases} 1 & \text{if} \quad v = \arg\min_j \|\mathbf{t}_i - \mathbf{c}_j\|_2 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

Note that all tokens share the same embedding space $\mathbf{C}$ which simplifies training.

We abuse $q(\mathbf{t}_i)$ to represent the index to the corresponding codebook entry. Then the quantized tokens $(\mathbf{c}_{q(\mathbf{t}_1)}, \mathbf{c}_{q(\mathbf{t}_2)}, \cdots, \mathbf{c}_{q(\mathbf{t}_M)})$ will be fed to the decoder network to recover the original pose:

$$\hat{\mathbf{G}} = f_d(\mathbf{c}_{q(\mathbf{t}_1)}, \mathbf{c}_{q(\mathbf{t}_2)}, \cdots, \mathbf{c}_{q(\mathbf{t}_M)}) \tag{3}$$

The network structure is similar to the encoder network in the reverse order except that we use a shallower MLP-Mixer network with only one block.

The encoder network, the codebook, and the decoder network are jointly learned by minimizing the following loss over the training dataset:

$$\ell_{pct} = \text{smooth}_{L_1}(\hat{\mathbf{G}}, \mathbf{G}) + \beta \sum_{i=1}^{M} \|\mathbf{t}_i - \text{sg}[\mathbf{c}_{q(\mathbf{t}_i)}]\|_2^2, \tag{4}$$

where, sg denotes stopping gradient, β is a hyperparameter.

We follow the optimization strategy used in [81] to handle the broken gradients issue in the quantization step and the codebook is updated using the exponential moving average of previous token features. In our implementation, we have two designs that improve the results. First, inspired by [26, 94], we randomly mask some joints and require the model to reconstruct them. Second, we concatenate the image features around the joints with the positional features to enhance its discrimination ability.

Discussion. We try to explain why PCT learns tokens that correspond to meaningful sub-structures of poses. At one extreme, if each token corresponds to a single joint, then we need $w \times h$ (*i.e.* 65536 for an image of size 256×256) codebook entries to achieve a small quantization error. But we only use 1024 entries in our experiments which is much smaller. This drives the model to learn larger structures than individual joints to improve the efficiency of the codebook. At another extreme, if we let a token correspond to an intact pose, then we only need one token instead of M tokens. But in the worst case, it requires $(wh)^K$ codebook entries in order to quantize the poses with a small error. In contrast, our method drives the model to divide a pose into multiple basic sub-structures whose possible configurations can be described by a shared set.

Relation to VQ-VAE [81]. The PCT representation is inspired by VQ-VAE. The main difference is that VQ-VAE treats well-defined regular data, *e.g.* image patches with the resolution of 16×16, as tokens. However, for human poses, we require PCT to automatically learn meaningful sub-structures as tokens, which is realized by the compositional encoder as well as the codebook sharing scheme. Besides, the network structures of the encoder and decoder are particularly designed for human poses, different from VQ-VAE.

3.2. Human Pose Estimation

With the learned codebook and the decoder, we cast human pose estimation as a classification task. As shown in Figure 3, given a cropped input image $\mathbf{I}$, we simply predict the categories of the M tokens, which are fed to the decoder to recover the pose. We use backbone for extracting image features $\mathbf{X}$ and design the following classification head.

Classification head. We first use two basic residual convolution blocks [28] to modulate the backbone features. Then, we flatten the features and change their dimension by a linear projection layer:

$$\mathbf{X}_f = \mathcal{L}(\text{Flatten}(\mathcal{C}(\mathbf{X}))), \tag{5}$$

where $\mathcal{C}$ and $\mathcal{L}$ represent the feature modulator and the linear projection respectively. We reshape the one-dimensional output feature into a matrix $\mathbf{X}_f \in \mathbb{R}^{M \times N}$, use four MLP-Mixer blocks [77] to process the features, and output the logits of token classification:

$$\hat{\mathbf{L}} = \mathcal{M}(\mathbf{X}_f), \tag{6}$$

where $\hat{\mathbf{L}}$ has the shape of $\mathbb{R}^{M \times V}$.

Training. We use two losses to train the classification head. First, we enforce the cross entropy loss:

$$\ell_{cls} = \mathrm{CE}(\hat{\mathbf{L}}, \mathbf{L}), \tag{7}$$

where $\mathbf{L}$ denotes the ground-truth token classes obtained by feeding the ground-truth poses into the encoder.

We also enforce a pose reconstruction loss, which minimizes the difference between the predicted and the ground-truth poses. To allow the gradients from the decoder network to flow back to the classification head, we replace the hard inference scheme with a soft version:

$$\mathbf{S} = \hat{\mathbf{L}} \times \mathbf{C}, \tag{8}$$

where $\mathbf{S} \in \mathbb{R}^{M \times N}$ denotes the linearly interpolated token features. The token features $\mathbf{S}$ are then fed to the pre-learned decoder to obtain the predicted pose $\hat{\mathbf{G}}$. The complete loss function is:

$$\ell_{all} = CE(\hat{\mathbf{L}}, \mathbf{L}) + \mathrm{smooth}_{L_1}(\hat{\mathbf{G}}, \mathbf{G}). \tag{9}$$

Note that the decoder network is not updated during training.

4. Experiments

We first extensively evaluate the PCT representation on five benchmark datasets in the context of 2D human pose estimation. Then we present the 3D pose estimation results and compare them to the state-of-the-art methods. Ablation studies about the main components of our method are also provided to help understand the approach.

4.1. Datasets and metrics

2D pose datasets. First, we conduct experiments on the COCO [42] and MPII [1] datasets. The COCO dataset has $150K$ labeled human instances for training, $5K$ images for validation, and $30K$ images for testing. The MPII dataset has $40K$ labeled human instances performing a variety of activities. Second, we evaluate our method on four datasets that have severe occlusions, including the test set of the CrowdPose [35] dataset, the validation and test sets of the OCHuman [107] dataset, and the SyncOCC [108] dataset. In CrowdPose [35] and OCHuman [107], the occluded joints are manually labeled by annotators. The SyncOCC [108] dataset is a synthetic dataset generated by UnrealCV [91] so it provides accurate locations of the occluded joints. We directly apply the model trained on the COCO dataset to the four datasets without re-training. We report the results on the occluded joints to validate the capability of the model to handle occlusion.

3D pose datasets. We conduct experiments on the Human3.6M [30] dataset which has 11 human subjects performing daily actions. We follow the practice of the previous works such as [17]. In particular, five subjects (S1, S5, S6, S7, S8) are used for training, and two subjects (S9, S11) are used for testing. Since there are no labels for joint occlusion, we only compare our method to the state-of-the-art methods to validate the general applicability of the representation to both 2D and 3D poses.

Evaluation metrics. We follow the standard evaluation metrics for the COCO [42], MPII [1] and, Human3.6M [30] datasets. In particular, the OKS-based AP (average precision), AP^{50} and AP^{75} are reported for the COCO dataset. The PCKh (head-normalized probability of correct keypoint) score is used for the MPII dataset. The MPJPE (mean per joint position error) are used for Human3.6M. On the four occlusion datasets, we report the AP^{OC} based on OKS computed only on the occluded joints.

4.2. Implementation details

We adopt the top-down estimation pipeline. In training, we use the GT boxes provided by the datasets. In testing, we use the detection results provided by [92] for COCO, and the GT boxes for MPII and the occlusion datasets following the common practice.

We use the Swin Transformer V2 [44, 45] backbone pre-trained with SimMIM [94] on ImageNet-1k [66]. It is also trained on the COCO dataset with heatmap supervision. To save computation cost, we fix the backbone and only train the classification head. We set the base learning rate, weight decay and batch size to $8e$-4, 0.05 and 256, respectively. In total, we train the head for 210 epochs on COCO and MPII, and 50 epochs on Human3.6M. The flip testing is used.

We use the default data augmentations provided by MMPose [18] including random scale (0.5, 1.5), random rotation $(-40°, 40°)$, random flip (50%), grid dropout and color jitter (h=0.2, s=0.4, c=0.4, b=0.4). We also add the half body augmentation for COCO. The image size is 256×256.

In learning the representation, we use the AdamW [46] optimizer with the base learning rate set to $1e$-2 and weight decay to 0.15, respectively. We warm up the learning rate for 500 iterations and drop the learning rate according to the cosine schedule. The batch size is 512. We train 50 epochs for 2D pose and 20 epochs for 3D pose.

4.3. Results on COCO, MPII and H36M

COCO. Table 1 shows the results of the state-of-the-art top-down pose estimation methods on COCO [42] test-dev2017 and COCO val2017 sets, respectively. For our method, we provide three models of different sizes. We can see that they achieve better or comparable accuracy as the other methods. For example, our smallest model with Swin-Base outperforms the previous dominant heatmap-based methods including HRNet [74], HRFormer [103], and TokenPose [40] with much faster inference speed. Similarly,

Table 1. Results on the COCO test-dev2017 and val2017 sets. The best results in the cited papers are reported. We set the batch size to 32 when testing the speed of all models on a single V100 GPU. Since the official pre-trained model of Swin [44] use square windows, we directly adopt the square input size to avoid domain gaps. While our input size seems larger than the competitors (*e.g.* 256 × 256 vs. 256 × 192), the number of valid pixels is almost the same because the additional regions are mostly padded meaningless pixels.

Method	Backbone	Input size	GFLOPs ↓	Speed (fps) ↑	COCO test-dev2017 ↑			COCO val2017 ↑		
					AP	AP^{50}	AP^{75}	AP	AP^{50}	AP^{75}
SimBa. [92]	ResNet-152	384 × 288	28.7	76.3	73.7	91.9	81.1	74.3	89.6	81.1
PRTR [36]	HRNet-W32	384 × 288	21.6	87.0	71.7	90.6	79.6	73.1	89.4	79.8
TransPose [97]	HRNet-W48	256 × 192	21.8	56.7	75.0	92.2	82.3	75.8	90.1	82.1
TokenPose [40]	HRNet-W48	256 × 192	22.1	52.9	75.9	92.3	83.4	75.8	90.3	82.5
HRNet [74, 86]	HRNet-W48	384 × 288	35.5	75.5	75.5	92.7	83.3	76.3	90.8	82.9
DARK [105]	HRNet-W48	384 × 288	35.5	62.1	76.2	92.5	83.6	76.8	90.6	83.2
UDP [29]	HRNet-W48	384 × 288	35.5	67.9	76.5	92.7	84.0	77.8	92.0	84.3
SimCC [39]	HRNet-W48	384 × 288	32.9	71.4	76.0	92.4	83.5	76.9	90.9	83.2
HRFormer [103]	HRFormer-B	384 × 288	29.1	25.2	76.2	92.7	83.8	77.2	91.0	83.6
ViTPose [96]	ViT-Base	256 × 192	17.9	113.5	75.1	92.5	83.1	75.8	90.7	83.2
ViTPose [96]	ViT-Large	256 × 192	59.8	40.5	77.3	93.1	85.3	78.3	91.4	85.2
ViTPose [96]	ViT-Huge	256 × 192	122.9	21.8	78.1	93.3	85.7	79.1	91.6	85.7
SimBa. [92]	Swin-Base	256 × 256	16.6	74.4	75.4	93.0	84.1	76.6	91.4	84.3
Our approach	Swin-Base	256 × 256	15.2	115.1	76.5	92.5	84.7	77.7	91.2	84.7
Our approach	Swin-Large	256 × 256	34.1	76.4	77.4	92.9	85.2	78.3	91.4	85.3
Our approach	Swin-Huge	256 × 256	118.2	31.7	78.3	92.9	85.9	79.3	91.5	85.9

Table 2. Results on the MPII [1] val set (PCKh@0.5).

Method	Hea.	Sho.	Elb.	Wri.	Hip.	Kne.	Ank.	Mean
SimBa. [92]	97.0	95.6	90.0	86.2	89.7	86.9	82.9	90.2
PRTR [36]	97.3	96.0	90.6	84.5	89.7	85.5	79.0	89.5
HRNet [74, 87]	97.1	95.9	90.3	86.4	89.1	87.1	83.3	90.3
DARK [105]	97.2	95.9	91.2	86.7	89.7	86.7	84.0	90.6
TokenPose [40]	97.1	95.9	90.4	86.0	89.3	87.1	82.5	90.2
SimCC [39]	97.2	96.0	90.4	85.6	89.5	85.8	81.8	90.0
Our (Swin-Base)	97.5	97.2	92.8	88.4	92.4	89.6	87.1	92.5

our largest model also achieves better results than the state-of-the-art ViTPose (huge) with $1.5x$ faster inference speed. The fast inference speed is mainly due to the fact that our method does not require any expensive post-processing.

MPII. The results on the MPII validation set are shown in Table 2. The image size is set to be 256 × 256 for all methods. Our approach significantly surpasses the other methods. Our approach gets better performance mainly for the joints on the lower body which are easier to be occluded by other objects. Compared to the other classification-based method SimCC [39], our method achieves an improvement of 2.5 under the metric of $\mathrm{PCKh@0.5}$.

H36M. It is straightforward to apply the PCT representation to 3D pose estimation. We first learn the encoder, the codebook and the decoder on the 3D poses. Then we train a classification head for 3D pose estimation. For simplicity, we directly use the backbone used in 2D pose estimation without re-training. The results are shown in Table 3. Our approach achieves a smaller error than the state-of-the-art monocular image-based methods. The results show that PCT is general and applies to both 2D and 3D poses.

Table 3. 3D pose estimation results on the Human3.6M [30] dataset. '*' means using extra 2D MPII [1] dataset for training. We report the MPJPE metric (mm). We only compare to the static image-based methods in the table.

Sharma *et al.* [70]	Zhao *et al.* [109]	Martinez *et al.* [51]	Moon *et al.* [52]	Liu *et al.* [43]	Xu and Takano [95]	Li *et al.* [37]	Gong *et al.* [24]	Zeng *et al.* [104]	*Sun *et al.* [75]	Zou and Tang [111]	*Li *et al.* [34]	Ours (Swin-Base)	Ours (Swin-Huge)
58.0	58.0	57.6	54.4	52.4	51.9	50.9	50.2	49.9	49.6	49.4	48.6	50.8	47.8

4.4. Results on CrowdPose, OCHuman, SyncOCC

We evaluate how our method performs in severe occlusions. The results on the four occlusion datasets are shown in Table 4. We can see that our PCT based approach significantly outperforms the other methods. Figure 5 shows some examples. There are several interesting observations. First, when a large portion of the human body is occluded, our method can predict a reasonable configuration for the occluded joints that is in harmony with the visible joints although there are no supporting visual features. This validates the strong context modeling capability of our method.

Table 4. The results of the state-of-the-art methods on the occlusion datasets. The numbers of the competitors are obtained by running their official models using the MMPose [18] framework. The metrics are computed only on the occluded joints that overlap with the COCO annotated joints. The GT bounding box is used. 'OC' denotes the OCHuman [107] dataset.

Method	Backbone	Input size	Speed (fps) ↑	2D Occluded Pose Estimation (AP^{OC} ↑)				
				OC-val [107]	OC-test [107]	CrowdPose [35]	SyncOCC [108]	SyncOCC-H [108]
HRNet [74, 86]	HRNet-W48	384 × 288	75.5	38.1	38.1	74.5	90.8	73.0
DARK [105]	HRNet-W48	384 × 288	62.1	38.6	39.2	74.9	91.2	73.8
UDP [29]	HRNet-W48	384 × 288	67.9	38.6	38.8	75.0	90.8	73.0
HRFormer [103]	HRFormer-B	384 × 288	25.2	40.5	40.3	72.4	91.9	75.7
Poseur [49]	HRFormer-B	384 × 288	25.8	44.4	45.6	73.9	93.1	78.5
ViTPose [96]	ViT-Huge	256 × 192	21.8	46.7	45.8	74.7	92.3	77.4
SimBa. [92]	Swin-Base	256 × 256	74.4	40.1	39.8	71.6	90.7	72.4
Our approach	Swin-Base	256 × 256	115.1	45.6	44.5	73.9	93.0	78.3
Our approach	Swin-Large	256 × 256	76.4	47.2	47.0	76.8	93.4	78.9
Our approach	Swin-Huge	256 × 256	31.7	**50.8**	**49.6**	**77.2**	**94.0**	**79.7**

Table 5. Comparison of four pose representations in a completely fair setting. We conduct the experiments with the Swin-Base and input size 256 × 256. The results are reported on the occluded joints that overlap with the COCO annotated joints. The GT bounding box is used. 'OC' denotes the OCHuman [107] dataset.

Method	OC-val	OC-test	CrowdPose	SyncOCC	SyncOCC-H
Heatmaps	40.1	39.8	71.6	90.7	72.4
Discrete Bins	40.5	39.9	71.9	91.1	73.6
Coordinates	41.5	41.5	72.7	91.9	75.7
Our PCT	**45.6**	**44.5**	**73.9**	**93.0**	**78.3**

Table 6. Ablation study of four main components: Compo (compositional design), MJM (masked joint modeling), IG (image guidance), and RecLoss (auxiliary pose reconstruction Loss). We report the AP^V for reconstructed poses, AP^P for predicted poses on the COCO val2017 set, and AP^{OC} on the SyncOCC test set. All results are obtained with the backbone Swin-Base and input size 256 × 256.

Compo	MJM	IG	RecLoss	AP^V	AP^P	AP^{OC}
				33.1	16.2	56.8
✓				98.9	65.5	88.3
✓	✓			99.0	72.7	91.2
✓	✓	✓		99.0	75.1	92.8
✓	✓	✓	✓	99.0	77.4	93.1

Second, when a small portion is occluded, our method can predict accurate positions based on the visual features in the neighborhood. For example, in the fourth example of the first row, the ankle joint of the rightmost person is correctly predicted based on the visual features of the legs. Third, it also shows stronger capability to resolve the ambiguities of other distracting persons.

We also compare the four representations including the coordinates, heatmaps, discrete bins, and PCT in a completely fair setting. The results are shown in Table 5. We can see that PCT achieves much better results than the dominant heatmap representation, leading by about 5.0 AP on OCHuman, 2.3 AP on SyncOCC, and 5.9 AP on the more challenging SyncOCC hard set.

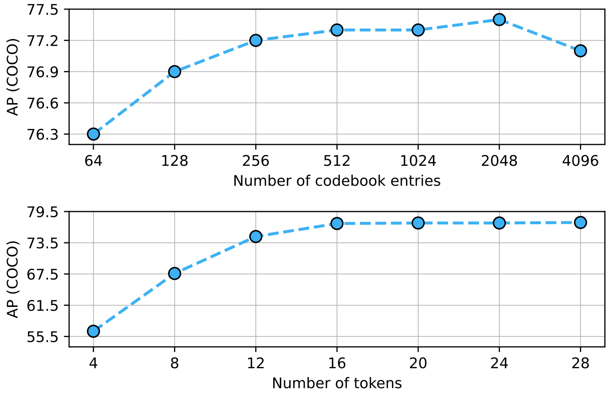

Figure 4. Impact of the number of codebook entries and the number of tokens, respectively. The results are obtained by the model using the Swin-Base backbone trained for 150 epochs on the COCO val2017 dataset.

4.5. Empirical analysis

Ablation study. We ablate the main components of PCT that we think are important. It includes the Compositional design (Compo), Masked Joints Modeling (MJM), Image Guidance (IG), and auxiliary Pose Reconstruction Loss (RecLoss). All experiments are conducted on the COCO val set and the SyncOCC set, using the Swin-Base backbone trained for 150 epochs.

The first baseline discards the compositional design and learns a codebook for each joint without interactions between the joints. As can be seen in Table 6, AP^V is only 33.1% meaning that the codebook cannot even reconstruct the poses accurately. This is because we need a significantly larger codebook without the compositional design. As a result, the pose estimation accuracy AP^P in the downstream task is only 16.2%. Adding the compositional design di-

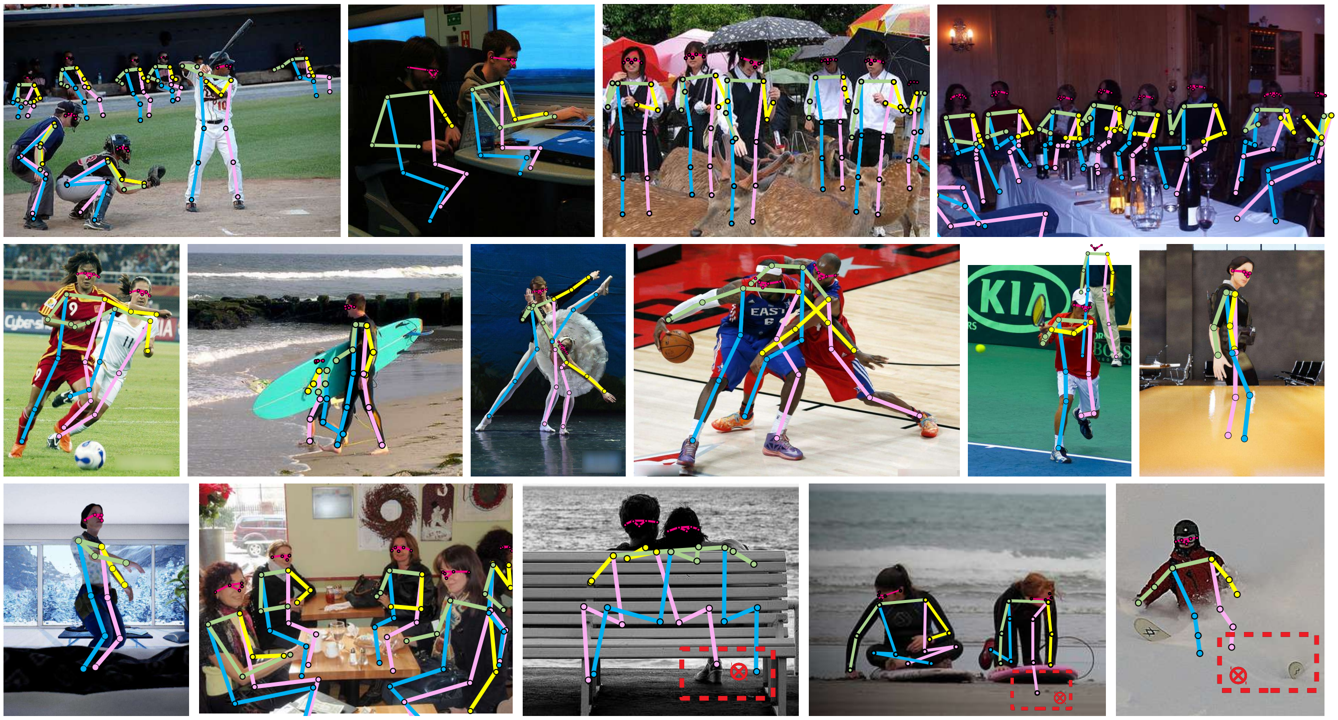

Figure 5. Qualitative results of our approach with Swin-Base backbone. The images are obtained from OChuman test set, COCO val2017 set, CrowdPose test set, and SyncOCC.

rectly improves AP^V to 98.9%. Adding MJM improves AP^P significantly from 65.5% to 72.7%. Our understanding is that MJM can drive the model to learn meaningful sub-structures (tokens) to help detect masked joints. IG and RecLoss also improve the results.

Token number. Increasing the number of tokens M will enlarge the representation space exponentially. The results are shown in Figure 4. We can see that increasing M from 4 to 16 notably improves the AP on the COCO dataset. Further increasing M brings little improvement. We find this is because the newly added tokens become redundant and have a large overlap with the existing ones. However, the results are barely affected by the redundant tokens which make the approach robust to the parameter.

Codebook size. Increasing the number of entries V in the codebook decreases the quantization error. However, it also increases the classification difficulty as the number of categories becomes larger. The results are shown in Figure 4. Setting this number between 256 and 2048 gives satisfactory results. Again, the model is not very sensitive to this parameter.

Qualitative results. Figure 5 shows some pose estimation results. We can see that it handles occlusion in a reasonable way. When a human body is occluded by a large region where even people are not completely sure about the exact pose, our method can predict a reasonable pose although it may be different from the GT pose. Note that they are not cherry-picked results. The last three examples show the failure cases. For the two people on the chair example, it is probable that the right ankle joint should be somewhere occluded by the chair. Similarly, for the person skating example, the ankle joints should be near the skateboard. The results suggest that leveraging objects as the context may further improve the estimation results.

5. Conclusion

In this work, we introduce a structured representation PCT to the human pose estimation community, which models the dependency between the body joints and automatically learns the sub-structures of the human pose. We also present a very simple pose estimation pipeline on top of the PCT representation, which does not need any complicated post-processing. It achieves better or comparable results as the state-of-the-art methods on five benchmarks. The discrete representation also paves the way for interacting with other discrete modalities such as text and speech.

Future work. It will be interesting to further reduce the ambiguities in pose estimation by exploring other cues under the discrete representation. For example, as mentioned in the qualitative study, we can model the context from the environments such as the surrounding objects.

References

[1] Mykhaylo Andriluka, Leonid Pishchulin, Peter V. Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *CVPR*, pages 3686–3693, 2014. 5, 6

[2] Mykhaylo Andriluka, Stefan Roth, and Bernt Schiele. Pictorial structures revisited: People detection and articulated pose estimation. In *2009 IEEE conference on computer vision and pattern recognition*, pages 1014–1021. IEEE, 2009. 1, 3

[3] Bruno Artacho and Andreas E. Savakis. Unipose: Unified human pose estimation in single images and videos. In *CVPR*, pages 7033–7042, 2020. 2

[4] Vasileios Belagiannis, Christian Rupprecht, Gustavo Carneiro, and Nassir Navab. Robust optimization for deep regression. In *ICCV*, pages 2830–2838, 2015. 2

[5] Irving Biederman, Robert J Mezzanotte, and Jan C Rabinowitz. Scene perception: Detecting and judging objects undergoing relational violations. *Cognitive psychology*, 14(2):143–177, 1982. 1

[6] Adrian Bulat and Georgios Tzimiropoulos. Human pose estimation via convolutional part heatmap regression. In *ECCV*, pages 717–732, 2016. 2

[7] Yuanhao Cai, Zhicheng Wang, Zhengxiong Luo, Binyi Yin, Angang Du, Haoqian Wang, Xiangyu Zhang, Xinyu Zhou, Erjin Zhou, and Jian Sun. Learning delicate local representations for multi-person pose estimation. In *ECCV*, pages 455–472, 2020. 2

[8] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7291–7299, 2017. 2

[9] João Carreira, Pulkit Agrawal, Katerina Fragkiadaki, and Jitendra Malik. Human pose estimation with iterative error feedback. In *CVPR*, pages 4733–4742, 2016. 2

[10] Ting Chen, Saurabh Saxena, Lala Li, Tsung-Yi Lin, David J. Fleet, and Geoffrey E. Hinton. A unified sequence interface for vision tasks. *CoRR*, abs/2206.07669, 2022. 3

[11] Yu Chen, Chunhua Shen, Xiu-Shen Wei, Lingqiao Liu, and Jian Yang. Adversarial posenet: A structure-aware convolutional network for human pose estimation. In *ICCV*, pages 1221–1230, 2017. 3

[12] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In *CVPR*, pages 7103–7112, 2018. 2

[13] Bowen Cheng, Bin Xiao, Jingdong Wang, Honghui Shi, Thomas S. Huang, and Lei Zhang. Higherhrnet: Scale-aware representation learning for bottom-up human pose estimation. In *CVPR*, 2020. 2

[14] Yu Cheng, Bo Wang, and Robby Tan. Dual networks based 3d multi-person pose estimation from monocular video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 1

[15] Yu Cheng, Bo Wang, Bo Yang, and Robby T Tan. Graph and temporal convolutional networks for 3d multi-person pose estimation in monocular videos. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1157–1165, 2021. 1

[16] Xiao Chu, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang. Structured feature learning for pose estimation. In *CVPR*, pages 4715–4723, 2016. 3

[17] Hai Ci, Chunyu Wang, Xiaoxuan Ma, and Yizhou Wang. Optimizing network structure for 3d human pose estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2262–2271, 2019. 5

[18] MMPose Contributors. Openmmlab pose estimation toolbox and benchmark. https://github.com/open-mmlab/mmpose, 2020. 5, 7

[19] Xiaochuan Fan, Kang Zheng, Yuewei Lin, and Song Wang. Combining local appearance and holistic view: Dual-source deep neural networks for human pose estimation. In *CVPR*, pages 1347–1355, 2015. 2

[20] Haoshu Fang, Shuqin Xie, Yu-Wing Tai, and Cewu Lu. RMPE: regional multi-person pose estimation. In *ICCV*, pages 2353–2362, 2017. 2

[21] Pedro F Felzenszwalb and Daniel P Huttenlocher. Pictorial structures for object recognition. *International journal of computer vision*, 61(1):55–79, 2005. 1, 3

[22] Mihai Fieraru, Anna Khoreva, Leonid Pishchulin, and Bernt Schiele. Learning to refine human pose estimation. In *CVPR*, pages 205–214, 2018. 2

[23] Zigang Geng, Ke Sun, Bin Xiao, Zhaoxiang Zhang, and Jingdong Wang. Bottom-up human pose estimation via disentangled keypoint regression. In *CVPR*, pages 14676–14686, 2021. 1, 2

[24] Kehong Gong, Jianfeng Zhang, and Jiashi Feng. Poseaug: A differentiable pose augmentation framework for 3d human pose estimation. In *CVPR*, pages 8575–8584, 2021. 6

[25] Kerui Gu, Linlin Yang, and Angela Yao. Removing the bias of integral pose regression. In *ICCV*, pages 11047–11056. IEEE, 2021. 2

[26] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. 4

[27] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. In *ICCV*, pages 2980–2988, 2017. 2

[28] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. 4

[29] Junjie Huang, Zheng Zhu, Feng Guo, and Guan Huang. The devil is in the details: Delving into unbiased data processing for human pose estimation. In *CVPR*, pages 5699–5708, 2020. 2, 6, 7

[30] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36(7):1325–1339, 2014. 5, 6

[31] Umar Iqbal, Pavlo Molchanov, and Jan Kautz. Weakly-supervised 3d human pose learning via multi-view images in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5243–5252, 2020. 1

[32] JongMok Kim, Hwijun Lee, Jaeseung Lim, Jongkeun Na, Nojun Kwak, and Jin Young Choi. Pose-mum: Reinforcing key points relationship for semi-supervised human pose estimation. *arXiv preprint arXiv:2203.07837*, 2022. 2

[33] Muhammed Kocabas, Salih Karagoz, and Emre Akbas. Multiposenet: Fast multi-person pose estimation using pose residual network. In *ECCV*, volume 11215, pages 437–453, 2018. 2

[34] Jiefeng Li, Siyuan Bian, Ailing Zeng, Can Wang, Bo Pang, Wentao Liu, and Cewu Lu. Human pose regression with residual log-likelihood estimation. In *ICCV*, 2021. 1, 2, 6

[35] Jiefeng Li, Can Wang, Hao Zhu, Yihuan Mao, Hao-Shu Fang, and Cewu Lu. Crowdpose: Efficient crowded scenes pose estimation and a new benchmark. In *CVPR*, 2019. 5, 7

[36] Ke Li, Shijie Wang, Xiang Zhang, Yifan Xu, Weijian Xu, and Zhuowen Tu. Pose recognition with cascade transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1944–1953, June 2021. 2, 6

[37] Shichao Li, Lei Ke, Kevin Pratama, Yu-Wing Tai, Chi-Keung Tang, and Kwang-Ting Cheng. Cascaded deep monocular 3d human pose estimation with evolutionary training data. In *CVPR*, pages 6172–6182. Computer Vision Foundation / IEEE, 2020. 6

[38] Wenbo Li, Zhicheng Wang, Binyi Yin, Qixiang Peng, Yuming Du, Tianzi Xiao, Gang Yu, Hongtao Lu, Yichen Wei, and Jian Sun. Rethinking on multi-stage networks for human pose estimation. *CoRR*, abs/1901.00148, 2019. 2

[39] Yanjie Li, Sen Yang, Shoukui Zhang, Zhicheng Wang, Wankou Yang, Shu-Tao Xia, and Erjin Zhou. Is 2d heatmap representation even necessary for human pose estimation?, 2021. 3, 6

[40] Yanjie Li, Shoukui Zhang, Zhicheng Wang, Sen Yang, Wankou Yang, Shu-Tao Xia, and Erjin Zhou. Tokenpose: Learning keypoint tokens for human pose estimation. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 1, 2, 3, 5, 6

[41] Ita Lifshitz, Ethan Fetaya, and Shimon Ullman. Human pose estimation using deep consensus voting. In *ECCV*, pages 246–260, 2016. 2

[42] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In *ECCV*, pages 740–755, 2014. 5

[43] Kenkun Liu, Rongqi Ding, Zhiming Zou, Le Wang, and Wei Tang. A comprehensive study of weight sharing in graph networks for 3d human pose estimation. In *ECCV*, volume 12355, pages 318–334, 2020. 6

[44] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, Furu Wei, and Baining Guo. Swin transformer V2: scaling up capacity and resolution. In *CVPR*, pages 11999–12009. IEEE, 2022. 5, 6

[45] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv preprint arXiv:2103.14030*, 2021. 5

[46] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. 5

[47] Jiasen Lu, Christopher Clark, Rowan Zellers, Roozbeh Mottaghi, and Aniruddha Kembhavi. Unified-io: A unified model for vision, language, and multi-modal tasks. *CoRR*, abs/2206.08916, 2022. 3

[48] Zhengxiong Luo, Zhicheng Wang, Yan Huang, Liang Wang, Tieniu Tan, and Erjin Zhou. Rethinking the heatmap regression for bottom-up human pose estimation. In *CVPR*, pages 13264–13273, 2021. 2

[49] Weian Mao, Yongtao Ge, Chunhua Shen, Zhi Tian, Xinlong Wang, Zhibin Wang, and Anton van den Hengel. Poseur: Direct human pose regression with transformers. October 2022. 2, 7

[50] Weian Mao, Zhi Tian, Xinlong Wang, and Chunhua Shen. Fcpose: Fully convolutional multi-person pose estimation with dynamic instance-aware convolutions. In *CVPR*, pages 9034–9043. Computer Vision Foundation / IEEE, 2021. 2

[51] Julieta Martinez, Rayat Hossain, Javier Romero, and James J. Little. A simple yet effective baseline for 3d human pose estimation. In *ICCV*, pages 2659–2668, 2017. 6

[52] Gyeongsik Moon, Ju Yong Chang, and Kyoung Mu Lee. Camera distance-aware top-down approach for 3d multi-person pose estimation from a single RGB image. In *ICCV*, pages 10132–10141, 2019. 6

[53] Gyeongsik Moon, Ju Yong Chang, and Kyoung Mu Lee. Posefix: Model-agnostic general human pose refinement network. In *CVPR*, pages 7773–7781, 2019. 2

[54] Alejandro Newell, Zhiao Huang, and Jia Deng. Associative embedding: End-to-end learning for joint detection and grouping. In *NeurIPS*, pages 2274–2284, 2017. 2

[55] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *ECCV*, pages 483–499, 2016. 1, 2

[56] Xuecheng Nie, Jiashi Feng, Jianfeng Zhang, and Shuicheng Yan. Single-stage multi-person pose machines. In *ICCV*, 2019. 2

[57] Xuecheng Nie, Jiashi Feng, Yiming Zuo, and Shuicheng Yan. Human pose estimation with parsing induced learner. In *CVPR*, 2018. 2

[58] Aude Oliva and Antonio Torralba. The role of context in object recognition. *Trends in cognitive sciences*, 11(12):520–527, 2007. 1

[59] George Papandreou, Tyler Zhu, Liang-Chieh Chen, Spyros Gidaris, Jonathan Tompson, and Kevin Murphy. Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model. In *ECCV*, pages 282–299, 2018. 2

[60] George Papandreou, Tyler Zhu, Nori Kanazawa, Alexander Toshev, Jonathan Tompson, Chris Bregler, and Kevin Murphy. Towards accurate multi-person pose estimation in the wild. In *CVPR*, pages 3711–3719, 2017. 1, 2

[61] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7753–7762, 2019. 1

[62] Xi Peng, Zhiqiang Tang, Fei Yang, Rogério Schmidt Feris, and Dimitris N. Metaxas. Jointly optimize data augmentation and network training: Adversarial data augmentation in human pose estimation. In *CVPR*, pages 2226–2234, 2018. 2

[63] Leonid Pishchulin, Mykhaylo Andriluka, Peter V. Gehler, and Bernt Schiele. Poselet conditioned pictorial structures. In *CVPR*, pages 588–595, 2013. 3

[64] Lingteng Qiu, Xuanye Zhang, Yanran Li, Guanbin Li, Xiaojun Wu, Zixiang Xiong, Xiaoguang Han, and Shuguang Cui. Peeking into occluded joints: A novel framework for crowd pose estimation. In *ECCV*, pages 488–504, 2020. 3

[65] Deva Ramanan. Learning to parse images of articulated objects. NeurIPS, 2006. 1, 3

[66] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Fei-Fei Li. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 5

[67] Nitin Saini, Elia Bonetto, Eric Price, Aamir Ahmad, and Michael J Black. Airpose: Multi-view fusion network for aerial 3d human pose and shape estimation. *IEEE Robotics and Automation Letters*, 7(2):4805–4812, 2022. 1

[68] Luca Schmidtke, Athanasios Vlontzos, Simon Ellershaw, Anna Lukens, Tomoki Arichi, and Bernhard Kainz. Unsupervised human pose estimation through transforming shape templates. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2484–2494, 2021. 1

[69] Taiki Sekii. Pose proposal networks. In *ECCV*, 2018. 2

[70] Saurabh Sharma, Pavan Teja Varigonda, Prashast Bindal, Abhishek Sharma, and Arjun Jain. Monocular 3d human pose estimation by generation and ordinal ranking. In *ICCV*, pages 2325–2334. IEEE, 2019. 6

[71] Dahu Shi, Xing Wei, Liangqi Li, Ye Ren, and Wenming Tan. End-to-end multi-person pose estimation with transformers. In *CVPR*, pages 11059–11068. IEEE, 2022. 2

[72] Kai Su, Dongdong Yu, Zhenqi Xu, Xin Geng, and Changhu Wang. Multi-person pose estimation with enhanced channel-wise and spatial information. In *CVPR*, pages 5674–5682, 2019. 2

[73] Ke Sun, Cuiling Lan, Junliang Xing, Wenjun Zeng, Dong Liu, and Jingdong Wang. Human pose estimation using global and local normalization. In *ICCV*, pages 5600–5608, 2017. 2

[74] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *CVPR*, 2019. 1, 2, 5, 6, 7

[75] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *Proceedings of the European conference on computer vision (ECCV)*, pages 529–545, 2018. 1, 2, 6

[76] Zhi Tian, Hao Chen, and Chunhua Shen. Directpose: Direct end-to-end multi-person pose estimation. In *CoRR*, 2019. 2

[77] Ilya O. Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, and Alexey Dosovitskiy. Mlp-mixer: An all-mlp architecture for vision. In *NeurIPS 2021, December 6-14, 2021, virtual*, pages 24261–24272, 2021. 4

[78] Jonathan J Tompson, Arjun Jain, Yann LeCun, and Christoph Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. *Advances in neural information processing systems*, 27, 2014. 2, 3

[79] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. In *CVPR*, pages 1653–1660, 2014. 1, 2

[80] Hanyue Tu, Chunyu Wang, and Wenjun Zeng. Voxelpose: Towards multi-camera 3d human pose estimation in wild environment. In *European Conference on Computer Vision*, pages 197–212. Springer, 2020. 1

[81] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017. 4

[82] Ali Varamesh and Tinne Tuytelaars. Mixture dense regression for object detection and human pose estimation. In *CVPR*, 2020. 2

[83] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Neurips*, pages 5998–6008, 2017. 2

[84] Timo Von Marcard, Roberto Henschel, Michael J Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 601–617, 2018. 1

[85] Jian Wang, Xiang Long, Yuan Gao, Errui Ding, and Shilei Wen. Graph-pcnn: Two stage human pose estimation with graph pose refinement. In *ECCV*, pages 492–508, 2020. 1, 2, 3

[86] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao. Deep high-resolution representation learning for visual recognition. *TPAMI*. 6, 7

[87] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao. Deep high-resolution representation learning for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 1, 6

[88] Yihan Wang, Muyang Li, Han Cai, Wei-Ming Chen, and Song Han. Lite pose: Efficient architecture design for 2d human pose estimation. In *CVPR*, pages 13116–13126. IEEE, 2022. 2

[89] Fangyun Wei, Xiao Sun, Hongyang Li, Jingdong Wang, and Stephen Lin. Point-set anchors for object detection, instance segmentation and pose estimation. In *ECCV*, pages 527–544, 2020. 2

[90] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *CVPR*, pages 4724–4732, 2016. 1, 2

[91] Yi Zhang Siyuan Qiao Zihao Xiao Tae Soo Kim Yizhou Wang Alan Yuille Weichao Qiu, Fangwei Zhong. Unrealcv: Virtual worlds for computer vision. *ACM Multimedia Open Source Software Competition*, 2017. 5

[92] Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking. In *ECCV*, pages 472–487, 2018. 2, 5, 6, 7

[93] Rongchang Xie, Chunyu Wang, Wenjun Zeng, and Yizhou Wang. An empirical study of the collapsing problem in semi-supervised 2d human pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11240–11249, 2021. 1, 2

[94] Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. Simmim: a simple framework for masked image modeling. In *CVPR*, pages 9643–9653. IEEE, 2022. 4, 5

[95] Tianhan Xu and Wataru Takano. Graph stacked hourglass networks for 3d human pose estimation. In *CVPR*, pages 16105–16114. Computer Vision Foundation / IEEE, 2021. 6

[96] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. Vitpose: Simple vision transformer baselines for human pose estimation, 2022. 1, 6, 7

[97] Sen Yang, Zhibin Quan, Mu Nie, and Wankou Yang. Transpose: Keypoint localization via transformer. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 2, 6

[98] Wei Yang, Shuang Li, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang. Learning feature pyramids for human pose estimation. In *ICCV*, pages 1290–1299, 2017. 2

[99] Wei Yang, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang. End-to-end learning of deformable mixture of parts and deep convolutional neural networks for human pose estimation. In *CVPR*, pages 3073–3082, 2016. 3

[100] Yi Yang and Deva Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *CVPR*, pages 1385–1392, 2011. 3

[101] Yiding Yang, Zhou Ren, Haoxiang Li, Chunluan Zhou, Xinchao Wang, and Gang Hua. Learning dynamics via graph neural networks for human pose estimation and tracking. In *CVPR*, pages 8074–8084, 2021. 3

[102] Hongwei Yi, Chun-Hao P Huang, Dimitrios Tzionas, Muhammed Kocabas, Mohamed Hassan, Siyu Tang, Justus Thies, and Michael J Black. Human-aware object placement for visual environment reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3959–3970, 2022. 1

[103] Yuhui Yuan, Rao Fu, Lang Huang, Weihong Lin, Chao Zhang, Xilin Chen, and Jingdong Wang. Hrformer: High-resolution transformer for dense prediction. 2021. 2, 5, 6, 7

[104] Ailing Zeng, Xiao Sun, Fuyang Huang, Minhao Liu, Qiang Xu, and Stephen Lin. Srnet: Improving generalization in 3d human pose estimation with a split-and-recombine approach. In *ECCV*, volume 12359, pages 507–523, 2020. 6

[105] Feng Zhang, Xiatian Zhu, Hanbin Dai, Mao Ye, and Ce Zhu. Distribution-aware coordinate representation for human pose estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2, 6, 7

[106] Hong Zhang, Hao Ouyang, Shu Liu, Xiaojuan Qi, Xiaoyong Shen, Ruigang Yang, and Jiaya Jia. Human pose estimation with spatial contextual information. In *CoRR*, 2019. 3

[107] Song-Hai Zhang, Ruilong Li, Xin Dong, Paul L. Rosin, Zixi Cai, Xi Han, Dingcheng Yang, Haozhi Huang, and Shi-Min Hu. Pose2seg: Detection free human instance segmentation. In *CVPR*, pages 889–898. Computer Vision Foundation / IEEE, 2019. 5, 7

[108] Zhe Zhang, Chunyu Wang, Weichao Qiu, Wenhu Qin, and Wenjun Zeng. Adafuse: Adaptive multiview fusion for accurate human pose estimation in the wild. *IJCV*, pages 1–16, 2020. 5, 7

[109] Long Zhao, Xi Peng, Yu Tian, Mubbasir Kapadia, and Dimitris N Metaxas. Semantic graph convolutional networks for 3d human pose regression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3425–3435, 2019. 6

[110] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. In *CoRR*, 2019. 1, 2

[111] Zhiming Zou and Wei Tang. Modulated graph convolutional network for 3d human pose estimation. In *ICCV*, pages 11457–11467. IEEE, 2021. 6