Hourglass Tokenizer for Efficient Transformer-Based 3D Human Pose Estimation

Wenhao Li[1] Mengyuan Liu[1*] Hong Liu[1] Pichao Wang[2†] Jialun Cai[1] Nicu Sebe[3]
[1]National Key Laboratory of General Artificial Intelligence, Peking University, Shenzhen Graduate School
[2]Amazon Prime Video [3]University of Trento
{wenhaoli,liumengyuan,hongliu}@pku.edu.cn pichaowang@gmail.com
cjl@stu.pku.edu.cn niculae.sebe@unitn.it

2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) | 979-8-3503-5300-6/24/$31.00 ©2024 IEEE | DOI: 10.1109/CVPR52733.2024.00064

Abstract

Transformers have been successfully applied in the field of video-based 3D human pose estimation. However, the high computational costs of these video pose transformers (VPTs) make them impractical on resource-constrained devices. In this paper, we present a plug-and-play pruning-and-recovering framework, called Hourglass Tokenizer (HoT), for efficient transformer-based 3D human pose estimation from videos. Our HoT begins with pruning pose tokens of redundant frames and ends with recovering full-length tokens, resulting in a few pose tokens in the intermediate transformer blocks and thus improving the model efficiency. To effectively achieve this, we propose a token pruning cluster (TPC) that dynamically selects a few representative tokens with high semantic diversity while eliminating the redundancy of video frames. In addition, we develop a token recovering attention (TRA) to restore the detailed spatio-temporal information based on the selected tokens, thereby expanding the network output to the original full-length temporal resolution for fast inference. Extensive experiments on two benchmark datasets (i.e., Human3.6M and MPI-INF-3DHP) demonstrate that our method can achieve both high efficiency and estimation accuracy compared to the original VPT models. For instance, applying to MotionBERT and MixSTE on Human3.6M, our HoT can save nearly 50% FLOPs without sacrificing accuracy and nearly 40% FLOPs with only 0.2% accuracy drop, respectively. Code and models are available at https://github.com/NationalGAILab/HoT.

1. Introduction

3D human pose estimation (HPE) from videos has numerous applications, such as action recognition [22, 25, 38], human-robot interaction [11, 53], and computer animation [30]. Current video-based 3D HPE methods mainly follow the pipeline of 2D-to-3D pose lifting [2, 4, 13, 21, 43, 44, 47].

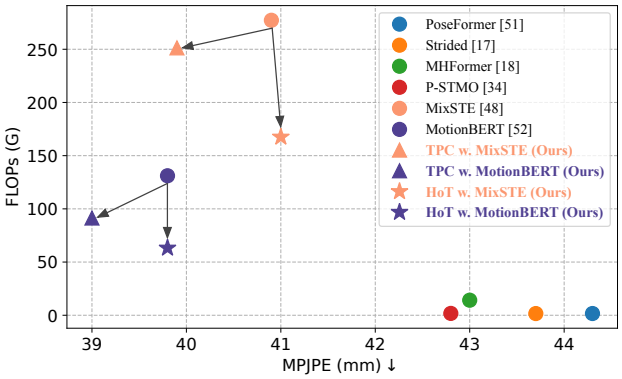


Figure 1. FLOPs and estimation errors (MPJPE, lower is better) of different VPTs on Human3.6M dataset. We achieve highly competitive or even better results while saving FLOPs.

This two-stage pipeline first utilizes an off-the-shelf 2D HPE model to detect 2D body joints for each video frame and then employs a separate lifting model to estimate 3D pose sequences from the detected 2D poses.

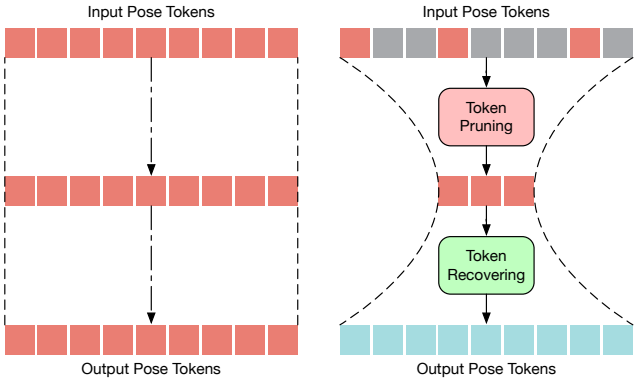
Recently, transformer-based architectures [18, 48, 51, 52] have shown state-of-the-art (SOTA) performance in the field of video-based 3D HPE, since they are effective at modeling the long-range dependencies among video frames. These video pose transformers (VPTs) typically regard each video frame as a pose token and utilize extremely long video sequences to achieve superior performance (*e.g.*, 81 frames in [51], 243 frames in [34, 48, 52], or 351 frames in [9, 17, 18]). However, these methods inevitably suffer from high computational demands since the VPT's self-attention complexity grows quadratically with respect to the number of tokens (*i.e.*, frames), hindering the deployment of these heavy VPTs on devices with limited computing resources.

To achieve efficient VPTs, two crucial factors require careful consideration: **(i)** Directly reducing the frame number can boost VPTs' efficiency, but it results in a small temporal receptive field that limits the model to capture richer spatio-temporal information to improve performance

*Corresponding Author.
†The work does not relate to author's position at Amazon.

2575-7075/24/$31.00 ©2024 IEEE
DOI 10.1109/CVPR52733.2024.00064

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:23:57 UTC from IEEE Xplore. Restrictions apply.



(a) VPT (b) HoT (Ours)

Figure 2. **(a)** Existing VPTs follow a "rectangle" paradigm that retains the full-length sequence across all blocks, which incurs expensive and redundant computational costs. **(b)** Instead, our HoT follows an "hourglass" paradigm that prunes the pose tokens and recovers the full-length tokens, which keeps a few tokens in the intermediate transformer blocks and thus improves the model efficiency. The gray squares represent the pruned tokens.

[23, 32]. Hence, it is essential to design an efficient solution while maintaining a large temporal receptive field for accurate estimation. **(ii)** Adjacent frames in a video sequence contain redundant information due to the similarity of nearby poses (50 Hz cameras used in Human3.6M [14]). Moreover, recent studies [16, 33, 40] found that some tokens tend to be similar in the deep transformer blocks. Thus, we infer that using full-length pose tokens in these blocks leads to redundant calculations and contributes little to the final estimation.

Based on these observations, we propose to prune pose tokens in the deep transformer blocks to improve the efficiency of VPTs. Although token pruning can reduce the number of tokens and bring efficiency, it also makes it difficult to estimate the consecutive 3D pose of all frames, as in existing VPTs [18, 48, 52], where each token corresponds to a frame. Additionally, for efficient inference, a real-world 3D HPE system should be able to estimate the 3D poses of all frames at once in an input video. Therefore, in order to make our method more compatible with being plugged into existing VPTs and achieve fast inference, we need to recover the original full-length tokens for all-frame estimation.

Driven by this analysis, we present a novel *pruning-and-recovering* framework for efficient transformer-based 3D HPE from videos. Different from existing VPTs that maintain the full-length sequence across all blocks, our method begins with pruning the pose tokens of redundant frames and ends with recovering the full-length tokens. By using these two designs, we can keep only a few tokens in the intermediate transformer blocks and thus improve the model efficiency (see Figure 2). For this to be achieved effectively, we argue that the key is to select a few representative tokens with high semantic diversity, as such tokens can maintain rich information while reducing video redundancy. Since the cluster centers can retain the semantic diversity of the original signal, we propose a token pruning cluster (TPC) module that utilizes the cluster to dynamically select the cluster centers as the representative tokens. Furthermore, we develop a lightweight token recovering attention (TRA) module to restore the detailed spatio-temporal information based on the selected tokens, which expands the low temporal resolution caused by pruning operation to the full temporal resolution. This strategy enables the network to estimate consecutive 3D poses of all frames, which facilitates fast inference.

Our method can be easily integrated into existing VPTs [18, 48, 52] with minimal modifications (see Figure 3). Specifically, the first few transformer blocks of VPTs remain unchanged to obtain pose tokens with comprehensive information from full video frames. These pose tokens are then pruned by our TPC, and the remaining tokens that serve as the representative tokens are further fed into the subsequent transformer blocks. Finally, the full-length tokens are recovered by TRA, which is added after the last transformer block, while the intermediate transformer blocks still use representative tokens. Thus the additional parameters and FLOPs from TRA are negligible. Since the number of tokens first decreases through pruning and then increases through recovering, we refer to the framework as an hourglass [31] and name it as Hourglass Tokenizer (HoT).

To validate the effectiveness and efficiency of our method, we deploy it on top of SOTA VPTs (MHFormer [18], MixSTE [48], and MotionBERT [52]). Extensive experiments demonstrate that existing VPTs consume huge unnecessary computational costs in capturing temporal information, and the proposed method can not only maintain the ability of the model but also reduce the computational costs. As shown in Figure 1, our HoT can reduce nearly 50% floating-point operations (FLOPs) on MotionBERT [52] without sacrificing performance and nearly 40% FLOPs on MixSTE [48] with only 0.2% performance loss.

The contributions of our paper are summarized below:

- We present HoT, a plug-and-play *pruning-and-recovering* framework for efficient transformer-based 3D HPE from videos. Our HoT reveals that maintaining the full-length pose sequence is redundant, and a few pose tokens of representative frames can achieve both high efficiency and performance.
- To accelerate VPTs effectively, we propose a TPC module to select a few representative tokens for video redundancy reduction and a TRA module to restore the original temporal resolution for fast inference.
- Extensive experiments conducted on three recent VPTs show that HoT achieves highly competitive or even superior results while significantly improving efficiency.

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:23:57 UTC from IEEE Xplore. Restrictions apply.

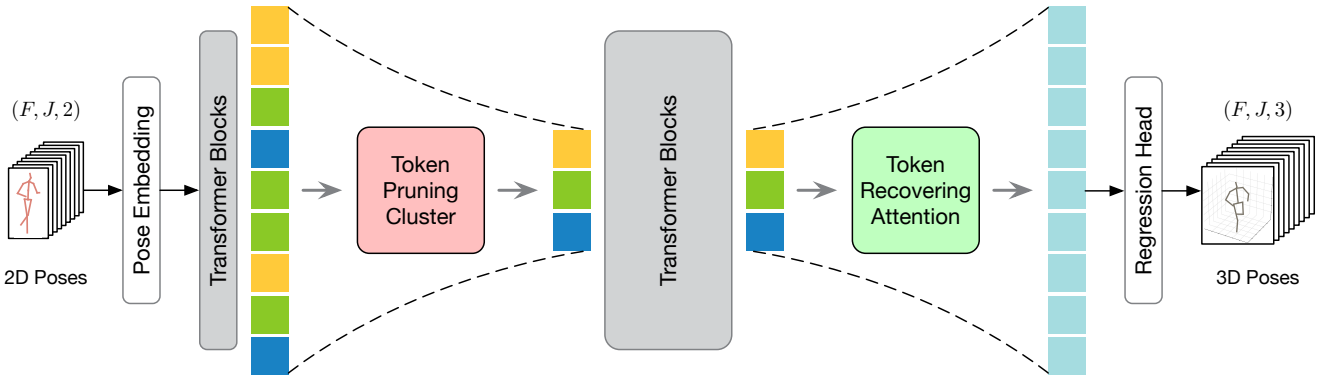


Figure 3. Overview of the proposed Hourglass Tokenizer (HoT). It mainly consists of a token pruning cluster (TPC) module and a token recovering attention (TRA) module. TPC selects the pose tokens of representative frames after the first few transformer blocks and TRA recovers the full-length tokens after the last transformer block.

2. Related Work

Transformer-based 3D HPE. Transformers are firstly proposed in [36] and have been successfully applied to video-based 3D HPE [18, 48, 51, 52]. These video pose transformers (VPTs) are often built to capture spatial and temporal information for 3D HPE using transformers. For instance, MHFormer [18] learns spatio-temporal multi-hypothesis representations of 3D human poses via transformers. MixSTE [48] proposes a mixed spatio-temporal transformer to capture the temporal motion of different body joints. MotionBERT [52] presents a dual-stream spatio-temporal transformer to model long-range spatio-temporal relationships among skeletal joints. However, the improved performance of these VPTs comes with a heavy computation burden.

Efficient 3D HPE. Efficient 3D HPE is critical in computing resource-constrained environments. Existing explorations mainly focus on efficient architecture design [6, 28, 32] and data redundancy reduction [9, 17, 34, 45]. VPose [32] presents a fully convolutional architecture that processes multiple frames in parallel. Strided [17] designs a strided transformer encoder to aggregate redundant sequences. Recently, several studies [9, 34, 45] have attempted to improve model efficiency by uniformly sampling video sequences. For example, DeciWatch [45] proposes a flow that takes sparsely sampled frames as inputs. However, this is sub-optimal as it simply selects frames at a fixed interval in a static manner without considering their contextual cues. In contrast, we propose to utilize the cluster to dynamically select pose tokens of representative frames with high-level semantic representations. Besides, many efficient methods [9, 17, 50] are designed for a specific model and none of them unifies the efficient design for different VPTs. We are the first to propose a plug-and-play framework for efficient VPTs, which can be plugged into common VPT models.

Token Pruning for Transformers. The self-attention complexity in transformers grows quadratically with the number of tokens, making it infeasible for high spatial or temporal resolution inputs. Many works [3, 7, 15, 24, 41] attempt to alleviate this issue by using token pruning, which aims to select significant tokens from different inputs. They find that discarding less informative tokens in the deep transformer blocks only leads to a slight performance drop. DynamicViT [33] proposes a learnable prediction module to estimate the scores of tokens and prune redundant tokens. PPT [26] selects important tokens based on the attention score. TCFormer [46] presents a token clustering transformer to cluster and merge tokens. In this work, we are the first to perform token pruning in VPTs for model acceleration. Unlike these studies that aim to reduce less related information (*e.g.*, image background) from images in the spatial domain, we focus on reducing temporal redundancy by selecting a few pose tokens of representative frames in the temporal domain. Furthermore, we propose to restore the full-length temporal resolution to meet the domain-specific requirement of efficient video-based 3D HPE.

3. Method

Figure 3 illustrates the overview of our Hourglass Tokenizer (HoT). Our HoT is a general-purpose *pruning-and-recovering* framework that can use different token pruning and token recovering strategies (see Sec 4.3). For better token pruning and recovering, we propose token pruning cluster (TPC) and token recovering attention (TRA) modules and insert them into SOTA VPTs [18, 48, 52]. Specifically, TPC takes the full-length pose tokens $x_n \in \mathbb{R}^{F\times J\times C}$ of n-th transformer block as inputs and outputs a few representative tokens $\tilde{x} \in \mathbb{R}^{f\times J\times C}$ ($f \ll F$), where J, F, and f are the number of body joints, input frames, and representative tokens, respectively. Here, C denotes the feature dimension. TRA recovers the full-length tokens from the tokens of the last transformer block $x_L \in \mathbb{R}^{f\times J\times C}$, where L is the number of transformer blocks, resulting in recovered tokens

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:23:57 UTC from IEEE Xplore. Restrictions apply.

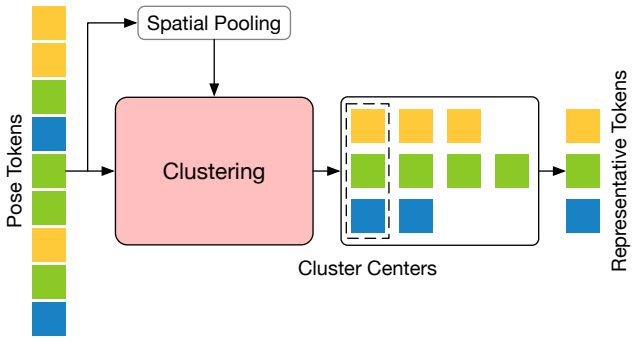


Figure 4. Illustration of our token pruning cluster (TPC) architecture. Given the input pose tokens, we pool them in the spatial dimension, cluster the input tokens into several groups according to the feature similarity of the resulting pooled tokens, and select the cluster centers as the representative tokens.

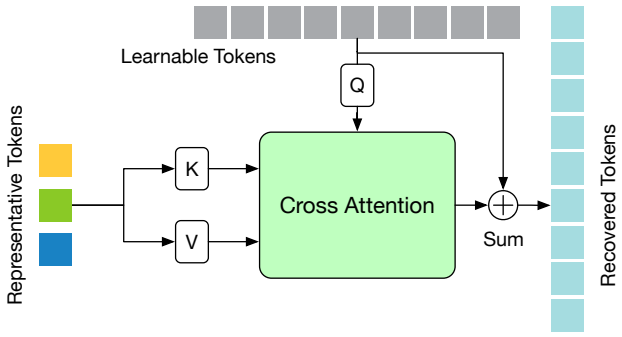


Figure 5. Illustration of our token recovering attention (TRA) architecture. TRA takes the representative tokens of the last transformer block, along with learnable tokens that are initialized to zero, as input to recover the full-length tokens.

$\hat{x} \in \mathbb{R}^{F\times J\times C}$. In the following section, we give details about the proposed TPC and TRA modules and show how to apply them to existing VPTs.

3.1. Token Pruning Cluster

We observe that the existing VPTs [18, 48, 52] take long video sequences as input and maintain the full-length sequence across all blocks (Figure 2 (a)), which is computationally expensive for high temporal resolution inputs. To tackle this issue, we propose to prune the pose tokens of video frames to improve the efficiency of VPTs. However, it is challenging to select a few pose tokens that maintain rich information for accurate 3D HPE.

To address this challenge, we propose a simple, effective, and parameter-free token pruning cluster (TPC) that dynamically selects a few pose tokens of representative frames to eliminate video redundancy. The architecture of TPC is illustrated in Figure 4. Given the input pose tokens of n-th transformer blocks $x_n \in \mathbb{R}^{F\times J\times C}$, an average spatial pooling is used along the spatial dimension to remove spatial redundancy, resulting in pooled tokens $\overline{x}_n \in \mathbb{R}^{F\times C}$. Then, we apply an efficient density peaks clustering based on k-nearest neighbors (DPC-kNN) algorithm [8]. This algorithm clusters the input pose tokens into several groups according to the feature similarity of the pooled tokens without requiring an iterative process.

The cluster centers of tokens are characterized by a higher density compared to their neighbors, as well as a relatively large distance from other tokens with higher densities. For a token $x^i \in \overline{x}_n$, the local density of tokens ρ is calculated by:

$$\rho_i = \exp(-\frac{1}{k}\sum_{x^j\in k\mathrm{NN}(x^i)} \left\|x^i - x^j\right\|_2^2), \qquad (1)$$

where $k\mathrm{NN}\left(x^i\right)$ are the k-nearest neighbors of a token x^i.

We then define the δ_i that measures the minimal distance between the token x^i and other tokens with higher density. The δ_i of the token with the highest density is set to the maximum distance between it and any other tokens. The δ_i of each token is calculated by:

$$\delta_i = \begin{cases} \min_{j:\rho_j>\rho_i} \left\|x^i - x^j\right\|_2, & \text{if } \exists \rho_j > \rho_i \\ \max_j \left\|x^i - x^j\right\|_2, & \text{otherwise} \end{cases}. \qquad (2)$$

The clustering center score of a token x^i is denoted by combining the local density ρ_i and minimal distance δ_i as $\rho_i \times \delta_i$. A higher score indicates that the token has both a large density and distance, showing a higher potential to be the cluster center. The top-f-scored input pose tokens are selected as cluster centers, and the remaining tokens are assigned to the nearest cluster center with higher density.

The cluster centers have high semantic diversity, containing more informative information than the other tokens. Therefore, the cluster centers serve as the representative tokens $\tilde{x} \in \mathbb{R}^{f\times J\times C}$ for efficient estimation, and the remaining tokens are discarded for reduction of video redundancy. Note that our method only prunes the tokens along the temporal dimension since the frame number F is much larger than the joint number J (*e.g.*, F=243 and J=17), *i.e.*, the expensive and redundant computational costs are dominated by the frame number in the temporal domain.

3.2. Token Recovering Attention

A large number of pose tokens have been pruned by TPC, which significantly reduces the computational costs. However, for fast inference, a real-world 3D HPE system should be capable of estimating the consecutive 3D poses of all frames in a given video (this is called *seq2seq* pipeline in [48]). Therefore, different from some token pruning methods in vision transformers that can use a few selected tokens to directly perform classification [20, 27, 33, 42], we need to recover the full-length tokens to keep the same number of tokens as the input video frames (in existing VPTs, each token corresponds to a frame). Meanwhile, for efficiency purposes, the recovering module should be lightweight.

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:23:57 UTC from IEEE Xplore. Restrictions apply.

To this end, a lightweight token recovering attention (TRA) module is proposed to restore the spatio-temporal information from the selected pose tokens, as shown in Figure 5. It only contains one multi-head cross-attention (MCA) layer without any additional networks. Formally, the dot-product attention [36] in the MCA is defined as:

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(QK^T/\sqrt{d}\right)V, \quad (3)$$

where queries $Q \in \mathbb{R}^{n_q \times d}$, keys $K \in \mathbb{R}^{n_k \times d}$, and values $V \in \mathbb{R}^{n_v \times d}$. d is the dimension and $\{n_q, n_k, n_v\}$ are the number of tokens for $\{Q, K, V\}$, respectively.

Our MCA takes the learnable tokens $x' \in \mathbb{R}^{F \times C}$ that are initialized to zero as queries and the j-th joint representative tokens of the last transformer block $x_L^j \in \mathbb{R}^{f \times C}$ as keys and values, followed by a residual connection:

$$\hat{x}^j = x' + \text{MCA}(x', x_L^j, x_L^j), \quad (4)$$

where $\text{MCA}(\cdot)$ is the function of MCA, and its inputs are queries, keys, and values. $\hat{x}^j \in \mathbb{R}^{F \times C}$ is the j-th joint recovered token, whose temporal dimension is the same as the queries (*i.e.*, the designed learnable tokens).

The TRA performs a reverse operation of selecting representative tokens, which recovers tokens of full-length temporal resolution from low ones using high-level spatio-temporal semantic information.

3.3. Applying to VPTs

Recent studies of VPTs can be divided into two types of pipelines based on their inference outputs: *seq2frame* [17, 18, 34, 51] and *seq2seq* [48, 52] pipelines. The *seq2frame* pipeline outputs the 3D pose of the center frame and requires repeated inputs of 2D pose sequences with significant overlap to predict the 3D poses of all frames. This pipeline can achieve better performance by considering both past and future information, but it is not efficient due to repeated calculations. In contrast, the *seq2seq* pipeline outputs 3D poses of all frames from the input 2D pose sequence at once, making it more efficient but leading to a degradation in performance. As a result, these two pipelines have their unique strengths, and we need to develop two strategies to better accommodate their different inference manners.

For the *seq2seq* pipeline, the outputs are all frames of the input video, and hence we need to restore the original temporal resolution. TPC and TRA are inserted into VPTs, where TPC prunes the tokens after a few transformer blocks and TRA recovers the full-length tokens after the last transformer block, as shown in Figure 3. Specifically, given the input 2D pose sequence $p \in \mathbb{R}^{F \times J \times 2}$ detected by an off-the-shelf 2D HPE detector from a video, we first feed them into a pose embedding module to embed spatial and temporal information of pose frames, resulting in tokens $x \in \mathbb{R}^{F \times J \times C}$. The embedded tokens are then fed into a few transformer blocks. Next, the TPC selects a few representative tokens $\tilde{x} \in \mathbb{R}^{f \times J \times C}$, which are the inputs of subsequent transformer blocks. After the last transformer block, the TRA restores the original temporal resolution and produces recovered tokens $\hat{x} \in \mathbb{R}^{F \times J \times C}$. Finally, a regression head is added to estimate the 3D pose sequence $q \in \mathbb{R}^{F \times J \times 3}$.

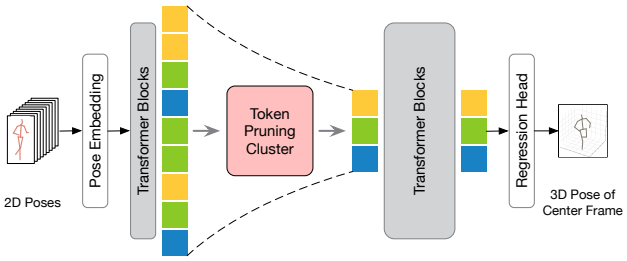


Figure 6. Illustration of our framework on *seq2frame* pipeline. The pose tokens are fed into TPC to select representative tokens. After the regression head, the 3D pose of the center frame is selected as the output for evaluation.

For the *seq2frame* pipeline, the output is the 3D pose of the center frame. Therefore, TRA is unnecessary and we only insert TPC into VPTs. Since the token of the center frame directly corresponds to the output and can provide crucial information to the final estimation, we concatenate it with the selected tokens to make this pipeline work better. As shown in Figure 6, the early stages of both pipelines share the same workflow. After the last transformer block, the tokens are directly sent to the regression head to perform regression and the 3D pose of center frame $q_{center} \in \mathbb{R}^{1 \times J \times 3}$ is selected as the final prediction.

4. Experiments

4.1. Datasets and Evaluation Metrics

Datasets. We evaluate our method on two 3D HPE benchmark datasets: Human3.6M [14] and MPI-INF-3DHP [29]. Human3.6M is the most widely used dataset for 3D HPE. It consists of 3.6 million video frames recorded by four RGB cameras at 50 Hz in an indoor environment. This dataset includes 11 actors performing 15 daily actions. Following [12, 19, 49, 54], subjects S1, S5, S6, S7, S8 are used for training and subjects S9, S11 are used for testing. MPI-INF-3DHP is another popular 3D HPE dataset. This dataset contains 1.3 million frames collected in indoor and outdoor scenes. It is smaller than Human3.6M but more challenging due to its diverse scenes, viewpoints, and motions.

Evaluation Metrics. For Human3.6M, we use the most commonly used mean per joint position error (MPJPE) as the evaluation metric, which measures the average Euclidean distance between estimated and ground truth 3D joint coordinates in millimeters. For MPI-INF-3DHP, we follow previous works [18, 48, 51] to report metrics of MPJPE,

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:23:57 UTC from IEEE Xplore. Restrictions apply.

Table 1. Comparison of efficiency and accuracy between *seq2seq* (∗) and *seq2frame* (†) inference pipelines. Frame per second (FPS) was computed on a single GeForce RTX 3090 GPU.

Method	Param (M)	FLOPs (G)	FPS	MPJPE ↓
MixSTE [48] (∗)	33.78	277.25	10432	40.9
HoT w. MixSTE (∗)	35.00	167.52 (↓ **39.6%**)	15770 (↑ **51.2%**)	41.0
MixSTE [48] (†)	33.78	277.25	43	40.7
TPC w. MixSTE (†)	33.78	161.73 (↓ **41.7%**)	68 (↑ **58.1%**)	40.4
MotionBERT [52] (∗)	16.00	131.09	14638	39.8
HoT w. MotionBERT (∗)	16.35	63.21 (↓ **51.8%**)	25526 (↑ **74.4%**)	39.8
MotionBERT [52] (†)	16.00	131.09	60	39.5
TPC w. MotionBERT (†)	16.00	61.04 (↓ **53.4%**)	109 (↑ **81.7%**)	39.2

percentage of correct keypoint (PCK) with the threshold of 150mm, and area under curve (AUC).

4.2. Implementation Details

The network is implemented using the PyTorch framework on one consumer-level NVIDIA RTX 3090 GPU with 24G memory. Our method builds upon MHFormer [18], MixSTE [48], and MotionBERT [52] for their largest frame number (*i.e.*, F=351, 243, 243) models. For a speed-accuracy trade-off, by default, we set $\{F$=351, n=1, f=117$\}$ for MHFormer, $\{F$=243, n=3, f=81$\}$ for MixSTE, and $\{F$=243, n=1, f=81$\}$ for MotionBERT. Note that MHFormer is designed for *seq2frame* pipeline, so we only implement our TPC on it. MixSTE and MotionBERT are designed for *seq2seq* pipeline and can be implemented on both *seq2frame* (with TPC) and *seq2seq* (with HoT) pipelines.

4.3. Ablation Study

To validate the effectiveness of our method, we conduct extensive ablation studies on Human3.6M dataset.

Inference Pipeline. In Table 1, we compare the efficiency and accuracy between different inference pipelines (mentioned in Sec 3.3). We conduct experiments on MixSTE [48] and MotionBERT [52] because both are designed for *seq2seq* pipeline and can be evaluated on both *seq2frame* and *seq2seq* pipelines. As shown in the table, the *seq2frame* can achieve better estimation accuracy by taking advantage of past and future information but lower efficiency due to repeated computations, *e.g.*, 40.7mm vs. 40.9mm and 43 FPS vs. 10432 FPS for MixSTE (about 243× lower). As our TPC is parameter-free and TRA is lightweight, our method with TPC introduces no additional parameters, and HoT w. MotionBERT only introduces additional 0.35M (2.2%) parameters, which can be neglected. Moreover, our method reduces the computational costs and improves the inference speed of these two pipelines, while maintaining or obtaining better performance.

For the *seq2seq*, our method can reduce the FLOPs of MixSTE and MotionBERT by 39.6% and 51.8% and improve the FPS by 51.2% and 74.4%, while estimation errors only drop 0.1mm (0.24%) and remain unchanged, respectively. For the *seq2frame*, our TPC w. MixSTE can reduce the FLOPs by 41.7% and improve the FPS by 58.1%, while bringing 0.3mm improvement. Additionally, our TPC w. MotionBERT can reduce 53.4% FLOPs and improve 81.7% FPS, while the estimation errors are reduced from 39.5mm to 39.2mm. Note that our method with TPC outperforms the one utilizing HoT. This is reasonable since our TRA in HoT is a reverse operation that uses inadequate information to recover the full-length tokens. In the following ablations, we take these two inference pipelines into account to sufficiently explore the proposed method, and we choose MixSTE [48] as the baseline since it is the first *seq2seq* transformer-based architecture and MotionBERT [52] is its follow-ups.

Table 2. Ablation study on the block index of representative tokens (n) under the *seq2frame* pipeline. Here, * denotes the result without re-training.

Method	Param (M)	FLOPs (G)	MPJPE*	MPJPE ↓
MixSTE [48]	33.78	277.25	40.7	40.7
TPC w. MixSTE, n=2	33.78	121.52 (↓ **56.2%**)	41.2	40.7
TPC w. MixSTE, n=3	33.78	147.47 (↓ **46.8%**)	41.2	40.5
TPC w. MixSTE, n=5	33.78	199.38 (↓ **28.1%**)	40.9	40.2
TPC w. MixSTE, n=7	33.78	251.29 (↓ **09.4%**)	**40.7**	**39.9**

Table 3. Ablation study on the number of representative tokens (f) under the *seq2seq* pipeline.

Method	Param (M)	FLOPs (G)	MPJPE ↓
MixSTE [48]	33.78	277.25	40.9
HoT w. MixSTE, f=9	34.96	114.90 (↓ **58.6%**)	43.5
HoT w. MixSTE, f=16	34.97	120.01 (↓ **56.7%**)	42.2
HoT w. MixSTE, f=61	34.99	152.90 (↓ **44.9%**)	41.2
HoT w. MixSTE, f=81	35.00	167.52 (↓ **39.6%**)	**41.0**
HoT w. MixSTE, f=135	35.03	206.98 (↓ **25.3%**)	41.3

Block index of Representative Tokens. The TPC can be inserted into optional transformer blocks, thereby adjusting the trade-off between computational costs and performance on demand in a flexible manner. Table 2 studies this under *seq2frame* pipeline (f is fixed to 61). Since TPC is a data-dependent scheme that introduces no extra parameters and transformers are input agnostic [1, 10], we can evaluate models with or without re-training. Increasing the block index of representative tokens can reduce the estimation error, but it also leads to higher computational costs. This indicates the deeper blocks of transformers contain more redundancy while the shallower blocks retain more useful information. Our method achieves competitive results without re-training while reducing FLOPs. When it works with re-training (training from scratch without pre-trained models), our method attains better performance. Our TPC w. MixSTE (n=2) achieves the same results while reducing 56.2% FLOPs and TPC w. MixSTE (n=7) improves the performance from 40.7mm to 39.9mm while reducing 9.4% FLOPs.

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:23:57 UTC from IEEE Xplore. Restrictions apply.

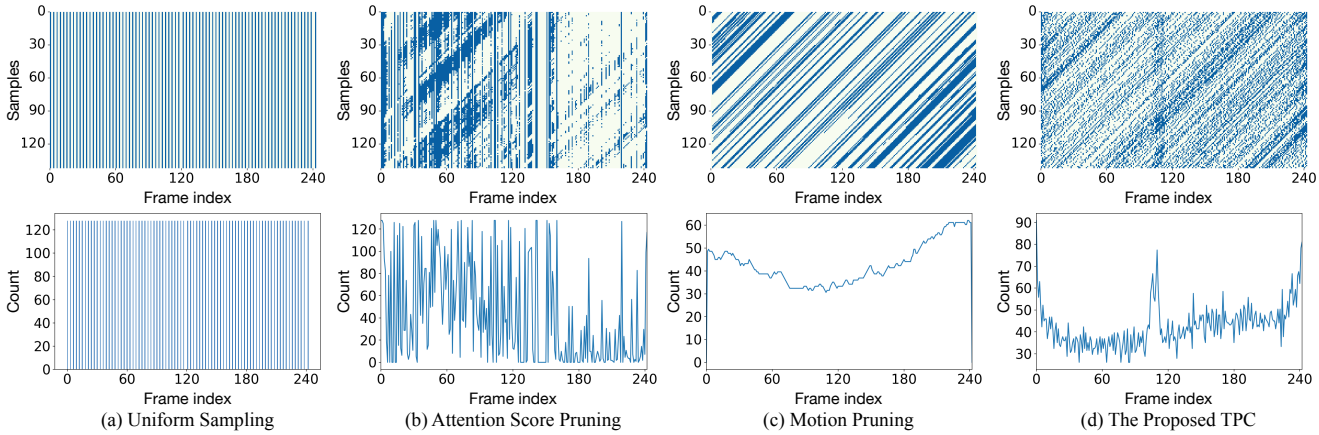


Figure 7. Statistics visualization of selected tokens for different token pruning strategies. **Top**: Frame indexes of selected tokens for some samples (140 samples) of consecutive video sequences (243 frames). Blue points are selected tokens and white points are pruned tokens. **Bottom**: Frequency count of frame indexes of selected tokens for these samples.

Table 4. Ablation study on the design choices of token pruning. "FN" denotes the frame noise that calculates the MPJPE of selected frames. "Full", "Pruned", "Selected", and "Center" denote the MPJPE of all frames, pruned frames, selected frames, and the center frame, respectively.

		seq2seq			*seq2frame*	
Method	FN	Full ↓	Pruned ↓	Selected ↓	Center ↓	Selected ↓
MixSTE [48]	6.61	40.9	-	-	40.7	-
Ours, Uniform Sampling	6.61	41.4	41.3	41.4	40.7	40.8
Ours, Attention Pruning	6.56	42.1	42.5	41.5	42.3	44.4
Ours, Motion Pruning	7.00	42.8	43.4	41.6	41.3	42.3
Ours, the Proposed TPC	6.63	**41.0**	**41.3**	**40.2**	**40.4**	**39.4**

Number of Representative Tokens. The number of representative tokens f can also be flexibly adjusted. In Table 3, we fix n to 3 and vary f under *seq2seq* pipeline. Decreasing f can reduce the FLOPs, but the best performance is achieved by using f=81. The reason for this is that an appropriate number of representative tokens can bring a good trade-off between retaining important information and reducing redundant information for both the pruning and recovering stages. Therefore, the optimal hyper-parameters for our HoT w. MixSTE are n=3 and f=81.

Token Pruning Design. Our HoT is a general-purpose *pruning-and-recovering* framework that can be equipped with different token pruning and recovering strategies. In Table 4, we compare different token pruning strategies, including attention pruning [26, 39], uniform sampling [9, 34, 45], and motion pruning that selects tokens with top-k-large motions. To measure the quality of selected tokens, we define a frame noise metric, which calculates the MPJPE of the 2D poses of input frames corresponding to the selected indexes. As the table shows, the frame noise values among these methods are similar (around 6.6mm) except for the motion pruning (7.0mm). This is because selecting tokens with top-k-large motion introduces some noise frames that differ significantly from clean frames, which can adversely affect

Table 5. Ablation study on the design choices of token recovering. Δ represents the performance gap between the results of pruned frames and selected frames.

Method	Param	FLOPs	Full ↓	Pruned ↓	Selected ↓	Δ
MixSTE [48]	33.78	277.25	40.9	-	-	-
Ours, Nearest Interpolation	33.78	161.73	41.5	42.2	40.2	2.0
Ours, Linear Interpolation	33.78	161.73	41.3	41.9	**40.0**	1.9
Ours, the Proposed TRA	35.00	167.52	**41.0**	**41.3**	40.2	**1.1**

performance. Moreover, our proposed TPC outperforms all other token pruning strategies, particularly for selected frames. Our TPC outperforms the uniform sampling strategy by 1.2mm (40.2mm vs. 41.4mm) and 1.4mm (39.4mm vs. 40.8mm) for selected frames under *seq2seq* and *seq2frame* pipelines, respectively. This emphasizes that the 3D pose results of our selected frames are easier to estimate, and our method can select more representative frames from a video.

Furthermore, we statistically visualize selected tokens of these four token pruning strategies. For better observation, we take samples of consecutive video sequences as input with a temporal interval of 1 between neighboring samples. The frame indexes and the frequency count of frame indexes of the selected tokens are shown in Figure 7 (top) and Figure 7 (bottom). Uniform sampling and motion pruning are static pruning methods because the former selects tokens at a fixed frame interval (equidistance in the top of Figure 7 (a)), while the latter selects tokens with the top-k-large motions that move with the input sequence (oblique triangle in the top of Figure 7 (c)). Instead, the attention score pruning and our method are dynamic methods that consider the significance of input tokens. The bottom of Figure 7 (b) shows that attention score pruning tends to select tokens in the left half of a sequence, indicating that the selected tokens tend to be similar to each other [40] and thus lack diversity. Our method primarily selects tokens at the beginning, center, and end of

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:23:57 UTC from IEEE Xplore. Restrictions apply.

Table 6. Comparison of parameters (M), FLOPs (G), and MPJPE with SOTA VPTs on Human3.6M. Here, F denotes the number of input frames. * indicates our re-implementation.

Method	F	Param	FLOPs	MPJPE ↓
PoseFormer (ICCV'21) [51]	81	9.60	1.63	44.3
Strided (TMM'22) [17]	351	4.35	1.60	43.7
P-STMO (ECCV'22) [34]	243	7.01	1.74	42.8
STCFormer (CVPR'23) [35]	243	18.93	156.22	40.5
MHFormer (CVPR'22) [18]	351	31.52	14.15	43.0
TPC w. MHFormer (Ours)	351	31.52	8.22 (↓ **41.91%**)	43.0
MixSTE (CVPR'22) [48]	243	33.78	277.25	40.9
HoT w. MixSTE (Ours)	243	35.00	167.52 (↓ **39.6%**)	41.0
TPC w. MixSTE (Ours)	243	33.78	251.29 (↓ **09.4%**)	39.9
MotionBERT (ICCV'23) [52]	243	16.00	131.09	39.2
MotionBERT (ICCV'23) [52]*	243	16.00	131.09	39.8
HoT w. MotionBERT (Ours)	243	16.35	63.21 (↓ **51.8%**)	39.8
TPC w. MotionBERT (Ours)	243	16.00	91.38 (↓ **30.3%**)	**39.0**

a sequence (the bottom of Figure 7 (d)). This is reasonable since these three parts can represent the rough motion of an entire sequence, which contributes a lot to accurate estimation. These findings highlight that our method not only eliminates the redundancy of video frames but also selects tokens with high semantic diversity (the top of Figure 7 (d) appears to be irregular), thus selecting more representative pose tokens for more accurate estimation.

Token Recovering Design. The token recovering strategies in our HoT can also be designed in different manners, as studied in Table 5. It shows that linear and nearest interpolation operations are parameter-free and achieve competitive results due to data redundancy (*i.e.*, nearby poses are similar) on Human3.6M (captured by 50 Hz cameras). Our TRA achieves better performance while introducing negligible parameters and FLOPs. These results validate the effectiveness of the proposed TRA, highlighting the benefits of using high-level semantic information for pose token recovering. Besides, the experiments show that the proposed TRA achieves the lowest performance gap between the estimated 3D poses of pruned frames and selected frames. This further demonstrates the effectiveness of our TRA, which can recover more accurate results based on the limited information provided by the selected tokens.

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

Human3.6M. Current SOTA performance on Human3.6M is achieved by transformer-based architectures. We compare our method with them by adding it to three very recent VPTs: MHFormer [18], MixSTE [48], and MotionBERT [52]. These three models significantly outperform previous works at the cost of high computational complexity, thus we choose them as baselines to evaluate our method. The comparisons are shown in Figure 1 and Table 6. We report the results of TPC w. MixSTE with $\{n{=}7,\ f{=}61\}$ and TPC w. MotionBERT with $\{n{=}2,\ f{=}121\}$. As shown in the table, our method can reduce the computational costs of recent VPTs while maintaining the ability of the model. For example, our HoT w. MotionBERT saves 51.8% FLOPs while maintaining accuracy, and our TPC w. MotionBERT obtains better performance with 0.8mm improvements while reducing computational costs by 30.3% in FLOPs. These results demonstrate the effectiveness and efficiency of our method, while also revealing that existing VPTs incur redundant computational costs that contribute little to the estimation accuracy or even decrease the accuracy. In addition, our method can remove these unnecessary computational costs while achieving comparable or even superior performance.

Table 7. Quantitative comparison with SOTA methods on MPI-INF-3DHP.

Method	PCK ↑	AUC ↑	MPJPE ↓
VPose (CVPR'19) [32] ($F{=}81$)	86.0	51.9	84.0
UGCN (ECCV'20) [37] ($F{=}96$)	86.9	62.1	68.1
Anatomy3D (TCSVT'21) [5] ($F{=}81$)	87.9	54.0	78.8
PoseFormer (ICCV'21) [51] ($F{=}9$)	88.6	56.4	77.1
MHFormer (CVPR'22) [18] ($F{=}9$)	93.8	63.3	58.0
TPC w. MHFormer (Ours, $F{=}9$)	94.0	63.3	58.4
MixSTE (CVPR'22) [48] ($F{=}27$)	94.4	66.5	54.9
HoT w. MixSTE (Ours, $F{=}27$)	**94.8**	**66.5**	**53.2**

MPI-INF-3DHP. We further evaluate our method on MPI-INF-3DHP dataset in Table 7. For a fair comparison, following [18, 48], we implement our method on MHFormer with $\{F{=}9,\ n{=}1,\ f{=}3\}$ and MixSTE with $\{F{=}27,\ n{=}3,\ f{=}9\}$. It can be found that our method (TPC w. MHFormer and HoT w. MixSTE) achieves competitive performance, demonstrating the effectiveness of our method in both indoor and outdoor scenes. Besides, our method can also work well with a small temporal receptive field.

5. Conclusion

This paper presents Hourglass Tokenizer (HoT), a plug-and-play *pruning-and-recovering* framework for efficient transformer-based 3D human pose estimation from videos. Our method reveals that maintaining the full pose sequence is unnecessary, and using a few pose tokens of representative frames can achieve both high efficiency and estimation accuracy. Comprehensive experiments demonstrate that our method is compatible and general. It can be easily incorporated into common VPT models on both *seq2seq* and *seq2frame* pipelines while effectively accommodating various token pruning and recovery strategies, thereby highlighting its potential for using future ones. We hope HoT can enable the creation of stronger and faster VPTs.

Acknowledgements. This work was supported by the National Natural Science Foundation of China (No. 62073004), Natural Science Foundation of Shenzhen (No. JCYJ20230807120801002), and by the MUR PNRR project FAIR (PE00000013) funded by the NextGenerationEU.

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:23:57 UTC from IEEE Xplore. Restrictions apply.

References

[1] Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your ViT but faster. In *ICLR*, 2022. 6

[2] Yujun Cai, Liuhao Ge, Jun Liu, Jianfei Cai, Tat-Jen Cham, Junsong Yuan, and Nadia Magnenat Thalmann. Exploiting spatial-temporal relationships for 3D pose estimation via graph convolutional networks. In *ICCV*, pages 2272–2281, 2019. 1

[3] Shuning Chang, Pichao Wang, Ming Lin, Fan Wang, David Junhao Zhang, Rong Jin, and Mike Zheng Shou. Making vision transformers efficient from a token sparsification view. In *CVPR*, pages 6195–6205, 2023. 3

[4] Hanyuan Chen, Jun-Yan He, Wangmeng Xiang, Zhi-Qi Cheng, Wei Liu, Hanbing Liu, Bin Luo, Yifeng Geng, and Xuansong Xie. HDFormer: High-order directed transformer for 3D human pose estimation. In *IJCAI*, pages 581–589, 2023. 1

[5] Tianlang Chen, Chen Fang, Xiaohui Shen, Yiheng Zhu, Zhili Chen, and Jiebo Luo. Anatomy-aware 3D human pose estimation with bone-based pose decomposition. *IEEE TCSVT*, 32(1):198–209, 2021. 8

[6] Sangbum Choi, Seokeon Choi, and Changick Kim. MobileHumanPose: Toward real-time 3D human pose estimation in mobile devices. In *CVPR*, pages 2328–2338, 2021. 3

[7] Zhiyang Dou, Qingxuan Wu, Cheng Lin, Zeyu Cao, Qiangqiang Wu, Weilin Wan, Taku Komura, and Wenping Wang. TORE: Token reduction for efficient human mesh recovery with transformer. In *ICCV*, pages 15143–15155, 2023. 3

[8] Mingjing Du, Shifei Ding, and Hongjie Jia. Study on density peaks clustering based on k-nearest neighbors and principal component analysis. *Knowledge-Based Systems*, 99:135–145, 2016. 4

[9] Moritz Einfalt, Katja Ludwig, and Rainer Lienhart. Uplift and upsample: Efficient 3D human pose estimation with uplifting transformers. In *WACV*, pages 2903–2913, 2023. 1, 3, 7

[10] Mohsen Fayyaz, Soroush Abbasi Koohpayegani, Farnoush Rezaei Jafari, Sunando Sengupta, Hamid Reza Vaezi Joze, Eric Sommerlade, Hamed Pirsiavash, and Jürgen Gall. Adaptive token sampling for efficient vision transformers. In *ECCV*, pages 396–414, 2022. 6

[11] Mercedes Garcia-Salguero, Javier Gonzalez-Jimenez, and Francisco-Angel Moreno. Human 3D pose estimation with a tilting camera for social mobile robot interaction. *Sensors*, 19 (22):4943, 2019. 1

[12] Kehong Gong, Jianfeng Zhang, and Jiashi Feng. PoseAug: A differentiable pose augmentation framework for 3D human pose estimation. In *CVPR*, pages 8575–8584, 2021. 5

[13] Wenbo Hu, Changgong Zhang, Fangneng Zhan, Lei Zhang, and Tien-Tsin Wong. Conditional directed graph convolution for 3D human pose estimation. In *ACMMM*, pages 602–611, 2021. 1

[14] 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 TPAMI*, 36(7):1325–1339, 2013. 2, 5

[15] Zhenglun Kong, Peiyan Dong, Xiaolong Ma, Xin Meng, Wei Niu, Mengshu Sun, Xuan Shen, Geng Yuan, Bin Ren, Hao Tang, et al. SPViT: Enabling faster vision transformers via latency-aware soft token pruning. In *ECCV*, pages 620–640, 2022. 3

[16] Ling Li, David Thorsley, and Joseph Hassoun. Sait: Sparse vision transformers through adaptive token pruning. *arXiv preprint arXiv:2210.05832*, 2022. 2

[17] Wenhao Li, Hong Liu, Runwei Ding, Mengyuan Liu, Pichao Wang, and Wenming Yang. Exploiting temporal contexts with strided transformer for 3D human pose estimation. *IEEE TMM*, 25:1282–1293, 2022. 1, 3, 5, 8

[18] Wenhao Li, Hong Liu, Hao Tang, Pichao Wang, and Luc Van Gool. MHFormer: Multi-hypothesis transformer for 3D human pose estimation. In *CVPR*, pages 13147–13156, 2022. 1, 2, 3, 4, 5, 6, 8

[19] Wenhao Li, Hong Liu, Hao Tang, and Pichao Wang. Multi-hypothesis representation learning for transformer-based 3D human pose estimation. *PR*, 141:109631, 2023. 5

[20] Youwei Liang, Chongjian Ge, Zhan Tong, Yibing Song, Jue Wang, and Pengtao Xie. Not all patches are what you need: Expediting vision transformers via token reorganizations. In *ICLR*, 2022. 4

[21] Hanbing Liu, Jun-Yan He, Zhi-Qi Cheng, Wangmeng Xiang, Qize Yang, Wenhao Chai, Gaoang Wang, Xu Bao, Bin Luo, Yifeng Geng, et al. PoSynDA: Multi-hypothesis pose synthesis domain adaptation for robust 3D human pose estimation. In *ACM MM*, pages 5542–5551, 2023. 1

[22] Mengyuan Liu, Hong Liu, and Chen Chen. Enhanced skeleton visualization for view invariant human action recognition. *PR*, 68:346–362, 2017. 1

[23] Ruixu Liu, Ju Shen, He Wang, Chen Chen, Sen-ching Cheung, and Vijayan Asari. Attention mechanism exploits temporal contexts: Real-time 3D human pose reconstruction. In *CVPR*, pages 5064–5073, 2020. 2

[24] Sifan Long, Zhen Zhao, Jimin Pi, Shengsheng Wang, and Jingdong Wang. Beyond attentive tokens: Incorporating token importance and diversity for efficient vision transformers. In *CVPR*, pages 10334–10343, 2023. 3

[25] Diogo C Luvizon, David Picard, and Hedi Tabia. Multi-task deep learning for real-time 3D human pose estimation and action recognition. *IEEE TPAMI*, 43(8):2752–2764, 2020. 1

[26] Haoyu Ma, Zhe Wang, Yifei Chen, Deying Kong, Liangjian Chen, Xingwei Liu, Xiangyi Yan, Hao Tang, and Xiaohui Xie. PPT: Token-pruned pose transformer for monocular and multi-view human pose estimation. In *ECCV*, pages 424–442, 2022. 3, 7

[27] Dmitrii Marin, Jen-Hao Rick Chang, Anurag Ranjan, Anish Prabhu, Mohammad Rastegari, and Oncel Tuzel. Token pooling in vision transformers. *arXiv preprint arXiv:2110.03860*, 2021. 4

[28] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. A simple yet effective baseline for 3D human pose estimation. In *ICCV*, pages 2640–2649, 2017. 3

[29] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt.

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:23:57 UTC from IEEE Xplore. Restrictions apply.

Monocular 3D human pose estimation in the wild using improved CNN supervision. In *3DV*, pages 506–516, 2017. 5

[30] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. VNect: Real-time 3D human pose estimation with a single rgb camera. *ACM TOG*, 36(4):1–14, 2017. 1

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

[32] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 3D human pose estimation in video with temporal convolutions and semi-supervised training. In *CVPR*, pages 7753–7762, 2019. 2, 3, 8

[33] Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. DynamicViT: Efficient vision transformers with dynamic token sparsification. In *NeurIPS*, pages 13937–13949, 2021. 2, 3, 4

[34] Wenkang Shan, Zhenhua Liu, Xinfeng Zhang, Shanshe Wang, Siwei Ma, and Wen Gao. P-STMO: Pre-trained spatial temporal many-to-one model for 3D human pose estimation. In *ECCV*, 2022. 1, 3, 5, 7, 8

[35] Zhenhua Tang, Zhaofan Qiu, Yanbin Hao, Richang Hong, and Ting Yao. 3D human pose estimation with spatio-temporal criss-cross attention. In *CVPR*, pages 4790–4799, 2023. 8

[36] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, pages 5998–6008, 2017. 3, 5

[37] Jingbo Wang, Sijie Yan, Yuanjun Xiong, and Dahua Lin. Motion guided 3D pose estimation from videos. In *ECCV*, pages 764–780, 2020. 8

[38] Pichao Wang, Wanqing Li, Zhimin Gao, Chang Tang, and Philip O Ogunbona. Depth pooling based large-scale 3D action recognition with convolutional neural networks. *IEEE TMM*, 20(5):1051–1061, 2018. 1

[39] Pichao Wang, Xue Wang, Fan Wang, Ming Lin, Shuning Chang, Hao Li, and Rong Jin. KVT: K-nn attention for boosting vision transformers. In *ECCV*, pages 285–302, 2022. 7

[40] Zhenyu Wang, Hao Luo, Pichao Wang, Feng Ding, Fan Wang, and Hao Li. VTC-LFC: Vision transformer compression with low-frequency components. In *NeurIPS*, pages 13974–13988, 2022. 2, 7

[41] Yutong Xie, Jianpeng Zhang, Yong Xia, Anton van den Hengel, and Qi Wu. ClusTR: Exploring efficient self-attention via clustering for vision transformers. *arXiv preprint arXiv:2208.13138*, 2022. 3

[42] Hongxu Yin, Arash Vahdat, Jose M Alvarez, Arun Mallya, Jan Kautz, and Pavlo Molchanov. A-ViT: Adaptive tokens for efficient vision transformer. In *CVPR*, pages 10809–10818, 2022. 4

[43] Yingxuan You, Hong Liu, Ti Wang, Wenhao Li, Runwei Ding, and Xia Li. Co-evolution of pose and mesh for 3D human body estimation from video. In *ICCV*, pages 14963–14973, 2023. 1

[44] 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*, pages 507–523, 2020. 1

[45] Ailing Zeng, Xuan Ju, Lei Yang, Ruiyuan Gao, Xizhou Zhu, Bo Dai, and Qiang Xu. DeciWatch: A simple baseline for 10x efficient 2D and 3D pose estimation. In *ECCV*, pages 607–624, 2022. 3, 7

[46] Wang Zeng, Sheng Jin, Wentao Liu, Chen Qian, Ping Luo, Wanli Ouyang, and Xiaogang Wang. Not all tokens are equal: Human-centric visual analysis via token clustering transformer. In *CVPR*, pages 11101–11111, 2022. 3

[47] Jinlu Zhang, Yujin Chen, and Zhigang Tu. Uncertainty-aware 3D human pose estimation from monocular video. In *ACMMM*, pages 5102–5113, 2022. 1

[48] Jinlu Zhang, Zhigang Tu, Jianyu Yang, Yujin Chen, and Junsong Yuan. MixSTE: Seq2seq mixed spatio-temporal encoder for 3D human pose estimation in video. In *CVPR*, pages 13232–13242, 2022. 1, 2, 3, 4, 5, 6, 7, 8

[49] Long Zhao, Xi Peng, Yu Tian, Mubbasir Kapadia, and Dimitris N Metaxas. Semantic graph convolutional networks for 3D human pose regression. In *CVPR*, pages 3425–3435, 2019. 5

[50] Qitao Zhao, Ce Zheng, Mengyuan Liu, Pichao Wang, and Chen Chen. PoseFormerV2: Exploring frequency domain for efficient and robust 3D human pose estimation. In *CVPR*, pages 8877–8886, 2023. 3

[51] Ce Zheng, Sijie Zhu, Matias Mendieta, Taojiannan Yang, Chen Chen, and Zhengming Ding. 3D human pose estimation with spatial and temporal transformers. In *ICCV*, pages 11656–11665, 2021. 1, 3, 5, 8

[52] Wentao Zhu, Xiaoxuan Ma, Zhaoyang Liu, Libin Liu, Wayne Wu, and Yizhou Wang. MotionBERT: A unified perspective on learning human motion representations. In *ICCV*, pages 15085–15099, 2023. 1, 2, 3, 4, 5, 6, 8

[53] Christian Zimmermann, Tim Welschehold, Christian Dornhege, Wolfram Burgard, and Thomas Brox. 3D human pose estimation in rgbd images for robotic task learning. In *ICRA*, pages 1986–1992, 2018. 1

[54] Zhiming Zou and Wei Tang. Modulated graph convolutional network for 3D human pose estimation. In *ICCV*, pages 11477–11487, 2021. 5

Authorized licensed use limited to: Peking University. Downloaded on October 21,2026 at 07:23:57 UTC from IEEE Xplore. Restrictions apply.