APP: Adaptive Pose Pooling for 3D Human Pose Estimation from Videos

Jinyan Zhang
State Key Laboratory of General Artificial Intelligence, Peking University, Shenzhen Graduate School
Shenzhen, China
zhangjinyan@stu.pku.edu.cn

Mengyuan Liu*
State Key Laboratory of General Artificial Intelligence, Peking University, Shenzhen Graduate School
Shenzhen, China
nkliuyifang@gmail.com

Hong Liu
State Key Laboratory of General Artificial Intelligence, Peking University, Shenzhen Graduate School
Shenzhen, China
hongliu@pku.edu.cn

Guoquan Wang
State Key Laboratory of General Artificial Intelligence, Peking University, Shenzhen Graduate School
Shenzhen, China
guoquanwang@stu.pku.edu.cn

Wenhao Li
State Key Laboratory of General Artificial Intelligence, Peking University, Shenzhen Graduate School
Shenzhen, China
wenhaoli@pku.edu.cn

Abstract

Current advancements in 3D human pose estimation have attained notable success by converting 2D poses into their 3D counterparts. However, this approach is inherently influenced by the errors introduced by 2D pose detectors and overlooks the intrinsic spatial information embedded within RGB images. To address these challenges, we introduce a versatile module called Adaptive Pose Pooling (APP), which is compatible with many existing 2D-to-3D lifting models. The APP module includes three novel sub-modules: Pose-Aware Offsets Generation (PAOG), Pose-Aware Sampling (PAS), and Spatial Temporal Information Fusion (STIF). First, we extract latent features of the multi-frame lifting model. Then, a 2D pose detector is utilized to extract multi-level feature maps from the image. After that, PAOG generates offsets according to featuremaps. PAS uses offsets to sample featuremaps. Then, STIF can fuse PAS sampling features and latent features. This innovative design allows the APP module to simultaneously capture spatial and temporal information. We conduct comprehensive experiments on two widely used datasets: Human3.6M and MPI-INF-3DHP. Meanwhile, we employ various lifting models to demonstrate the efficacy of the APP module. Our results show that the proposed APP module consistently enhances the performance of lifting models, achieving state-of-the-art results. Significantly, our module achieves these performance boosts without necessitating alterations to the architecture of the lifting model. Our code is available at https://github.com/jinyanzhang/APP.

*Mengyuan Liu is with the State Key Laboratory of General Artificial Intelligence, Peking University, Shenzhen Graduate School. Corresponding author is Mengyuan Liu (e-mail: nkliuyifang@gmail.com).

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
MM '24, October 28-November 1, 2024, Melbourne, VIC, Australia
© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-0686-8/24/10
https://doi.org/10.1145/3664647.3680880

CCS Concepts

• **Computing methodologies** → **Activity recognition and understanding**.

Keywords

3D Pose Estimation, Adaptive Pose Pooling, Feature Fusion

ACM Reference Format:
Jinyan Zhang, Mengyuan Liu, Hong Liu, Guoquan Wang, and Wenhao Li. 2024. APP: Adaptive Pose Pooling for 3D Human Pose Estimation from Videos. In *Proceedings of the 32nd ACM International Conference on Multimedia (MM '24), October 28-November 1, 2024, Melbourne, VIC, Australia.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3664647.3680880

1 Introduction

3D human pose estimation (HPE) is a critical task in computer vision, and its goal is to estimate 3D human poses from images, videos, and 2D human poses. It attempts to estimate precise positions and orientations of 3D human keypoints. The applications of 3D HPE are pretty large, such as virtual/augmented reality [24, 29], motion analysis [2, 8, 43], human-computer interaction [1, 12], and autonomous driving [13] etc. Models for 3D HPE can be broadly categorized into two fundamental classes based on the number of viewpoints employed: multi-view and single-view. Multi-view approaches leverage information from multiple camera perspectives, as explored in studies such as [6, 14, 17, 34, 46]. They mitigate depth ambiguity and reach better results than monocular 3D HPE methods. However, their practical applicability is hindered by the necessity of employing multiple cameras.

In contrast, single-view models offer a more applicable alternative and can be subdivided into single-stage and two-stage methodologies. Single-stage methods [3, 18, 25, 31, 40, 50] directly infer 3D human pose from input images or video frames without intermediate steps. While two-stage like early methods [26, 32] adopt a sequential approach. As shown in Figure 1(a), off-the-shelf 2D pose detectors [5, 30, 39] extract 2D human poses from input images or video frames. After that, they can obtain corresponding 3D human poses based on these 2D estimations, so this stage is often called the lifting stage. Models in the lifting stage can further be

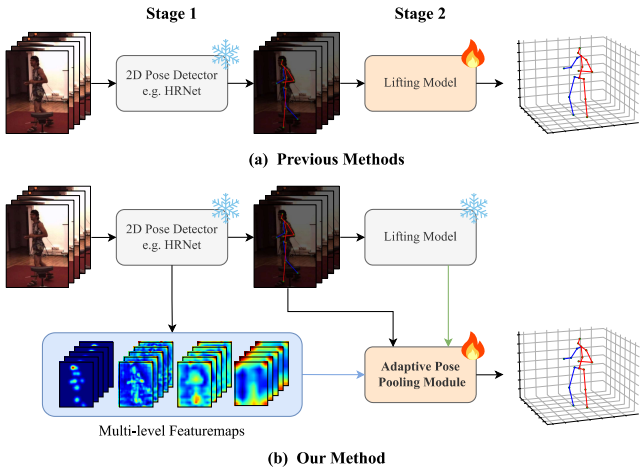


Figure 1: Difference between our method and previous methods. (a) Previous methods freeze the 2D pose detector and train the lifting model. (b) Our proposed method utilizes the 2D pose features extracted by the lifting model and takes multi-level feature maps extracted by the 2D pose detector as inputs. Our method addresses the issue of underutilizing image features by leveraging multi-level feature maps extracted by a pretrained 2D pose detector. Furthermore, while single-frame models utilize image features, they lack temporal information.

categorized into single-frame or multi-frame variants. Single-frame models [7, 47] infer 3D human poses from individual images, while multi-frame models process sequences of 2D human poses. The latter one contains two categories: many-to-one and many-to-many methods. Many-to-one models estimating the 3D human pose of a central frame using multiple 2D human poses [21, 22, 48, 49], and many-to-many, also called seq2seq models inferring every frame of a 3D human pose from corresponding 2D human poses [4, 27, 33, 37, 45, 51]. They try to address different problems, ranging from spatio-temporal information exploration to enhancing model robustness.

In this paper, we focus on monocular 3D HPE. It uses only one camera to obtain images of humans. Therefore, it introduces ambiguity because of the lack of depth information, which is an ill-posed task. Besides, occlusion, including self-occlusion and occlusion by other objects and persons, needs to be drawn to researchers' attention. Hence, single-view models sometimes fail to get accurate and robust 3D human poses owing to depth ambiguity and occlusion. **Many monocular 3D HPE models suffer from them because the information of the input images fails to be fully used.** HEMlets Pose [50] and Context-Aware PoseFormer (CA-PF) [47] try to handle this. HEMlets Pose explicitly lets the model learn relative depth among different keypoints in 2D human poses. In comparison, CA-PF adopts Deformable Attention [52] to extract features of feature maps obtained by the 2D pose detector. However, these methods [47, 50] are **single-frame models that cannot capture temporal dynamics**. Recently, multi-frame lifting models have reached significant improvement. **Why not use both image and temporal features?** This work proposes a well-designed Adaptive Pose Pooling (APP) module. Serving as a plug-and-play component that can be integrated with many existing lifting models. The APP module facilitates extracting temporal and spatial information from multi-frame 2D human pose data and feature maps. Figure 1 illustrates the distinction between previous methods and our proposed method. Previous methods freeze the 2D pose detector and only take detected 2D poses as the model input to estimate corresponding 3D poses. Our method uses feature maps extracted by a 2D pose detector and takes multi-frame lifting models' calculated pose features. Therefore, the APP module inherently can capture spatio-temporal features. **Do we really need to use all the features of the feature maps?** Features near 2D poses should be more important than features at other positions. To enhance the model's flexibility, the model should adjust the sampling points during training. In the meantime, getting spatial and temporal features and interacting among features is also important for the model.

Our proposed APP module contains three novel submodules: Pose-Aware Offsets Generation (PAOG), Pose-Aware Sampling (PAS), and Spatial Temporal Information Fusion (STIF). To address the problem of insufficient utilization of spatial information in images, we propose PAOG and PAS. PAOG leverages 2D human poses as indices to learn offsets during training. PAS adaptively extracts features from feature maps generated by the 2D human pose detector. Though single-frame models use image features, they suffer from lacking temporal features. We proposed STIF to solve this problem by fusing image and temporal features. We conduct extensive experiments on two popular datasets: Human3.6M and MPI-INF-3DHP. In the meantime, many previous models are selected as the baselines to demonstrate the efficacy of the proposed APP module. The experimental results show that our method achieves a new state-of-the-art performance even with different 2D pose detectors on Human3.6M, with 0.2mm (from 39.2mm to 39.0mm), 0.5mm (from 37.5mm to 37.0mm) and 0.4mm (from 36.6mm to 36.2mm) improvement in terms of Mean Per-Joint Position Error (MPJPE) for CPN[5], SH[30], HRNet[39], respectively. There are 4.6 (from 85.9 to 89.5) and 3.5mm (from 16.2mm to 12.7mm) improvements for Area Under Curve (AUC) and MPJPE on MPI-INF-3DHP.

Compared to current methods for monocular 3D HPE, our contributions are summarized as follows:

- We propose the APP module, enabling multi-frame lifting models to utilize information from the previous stage. This module is a plug-and-play solution compatible with most multi-frame lifting models and mitigates the problem of excessive reliance on 2D human pose outputs.
- We introduce Pose-Aware Offsets Generation (PAOG) and Pose-Aware Sampling (PAS). PAOG adjusts offsets relative to 2D human poses during training, allowing PAS to learn to adaptively capture spatial features from feature maps.
- We employ Spatial Temporal Information Fusion (STIF) to facilitate information interaction between the features sampled by PAS and the hidden feature of the lifting model. This enables the APP module to capture both temporal and spatial features simultaneously.
- We conduct experiments on two widely used datasets: Human3.6M and MPI-INF-3DHP. Our proposed approach

achieves state-of-the-art performance on both datasets compared to previous methods. Importantly, it does not alter the lifting model's architecture.

2 Related Work

Transformers[42] initially dominated the domain of natural language processing. Owing to its expressive performance, it has gained significant attention in computer vision as well [10]. Therefore, Transformer-based models have been increasingly applied across various domains, including 3D human pose estimation. PoseFormer [49] stands out as the pioneering model to integrate Transformers into 3D human pose estimation. Building upon this foundation, PoseFormerV2 [48] represents an advancement, introducing a frequency domain approach to enhance model robustness against fast motion changes. Furthermore, MHFormer [22] addresses the challenge of depth ambiguity by learning the representations of multiple hypothesis poses.

In the quest for improved efficiency and accuracy, novel architectural modifications have emerged. StridedTrans [21] utilizes fully connected layers in the Transformer encoder with stride convolutions. It reduces computational complexity and progressively shortens sequence lengths, thus enhancing pose estimation accuracy for the central frame. Meanwhile, MixSTE [45] focuses on modeling strong intra-frame correspondences between individual joints, thereby facilitating the learning of spatio-temporal correlations. P-STMO [37] adopts a strategy of occlusion masking to enable the model to learn more resilient pose representations. Additionally, it leverages a large volume of unlabeled data for self-supervised pretraining, thereby augmenting generalization capabilities. HoT [23] achieves this by implementing feature compression and restoration techniques, rendering it compatible with various lifting models while reducing computational overhead.

Advancements in attention mechanisms have also been pivotal. HDFormer [4], for instance, integrates self-attention and higher-order attention mechanisms to construct a hierarchical attention module aimed at mitigating complexities and handling heavily occluded scenarios. STCFormer [41] decomposes correlation learning into spatial and temporal dimensions, thereby reducing model complexity. Moreover, MotionBERT [51] emerges as a dual-stream spatio-temporal Transformer capable of concurrently capturing spatio-temporal information, showcasing state-of-the-art performance across multiple downstream tasks. Similarly, MotionAGFormer [27] introduces a hybrid architecture composed of multiple dual-stream modules, effectively capturing both spatial and temporal features of keypoints in tandem. KTPFormer[33] tries to overcome the weakness in existing transformer-based methods by proposing two plug-and-play attention modules, namely Kinematics Prior Attention (KPA) and Trajectory Prior Attention (TPA). These modules take advantage of the structure of the human body and motion trajectory information.

For the 2D-to-3D lifting stage, given a sequence of 2D human poses $\boldsymbol{p} \in \mathbb{R}^{T\times J\times C_{in}}$, the goal is to estimate the corresponding 3D human poses $\hat{\boldsymbol{p}} \in \mathbb{R}^{T\times J\times C_{out}}$, where T is the sequence length of 2D pose frames, J is the number of keypoints of the 2D human pose, C_{in} and C_{out} are the dimension of the 2D and 3D human pose keypoints, respectively.

These advancements collectively underscore the profound impact and ongoing evolution of Transformer-based approaches in advancing the field of 3D human pose estimation. Unlike our proposed method, which integrates image features into estimating 3D human poses, none of the existing transformer-based multi-frame methods take advantage of such information. This presents a unique opportunity for our approach to bridge this gap and potentially unlock new avenues for improving the accuracy and robustness of 3D human pose estimation methods.

3 Approach

3.1 Framework Overview

Figure 2 illustrates the overview of our proposed method. Given input images $\boldsymbol{I} \in \mathbb{R}^{t\times 3\times H\times W}$, where t is the number of image frames and H, W are the height and width of the image, note that $t \ll T$. We first use a 2D pose detector to extract the multi-level feature maps denoted by $\{\boldsymbol{H}_i\}_{s=1}^{S}$, where S is the number of scales. $\boldsymbol{H}_1$ is a heatmap to get the 2D pose p. Then pose features $\boldsymbol{P} \in \mathbb{R}^{T\times J\times d_{\boldsymbol{P}}}$ is calculated in the multi-frame lifting models without regression head, where $d_{\boldsymbol{P}}$ is the dimension of pose features. This process is formulated as:

$$\begin{aligned} \{\boldsymbol{H}_i\}_{s=1}^{S} &= Backbone(\boldsymbol{I}), \\ \boldsymbol{P} &= MLM(\boldsymbol{p}) \end{aligned} \tag{1}$$

$MLM(\cdot)$ denotes multi-frame lifting models without regression head; the backbone is a 2D pose detector. In Figure 2, the values of averaged image features $\bar{\boldsymbol{F}} \in \mathbb{R}^{T\times(k^2\times J)\times d}$ and hidden features $\boldsymbol{X} \in \mathbb{R}^{T\times J\times d}$ are updated by each layer of the APP module, where k is the window size.

To get the initialized offsets $\boldsymbol{\Delta}^0 \in \mathbb{R}^{T\times(k^2\times J)\times C_{in}}$, we repeat the values by taking each point in the 2D human poses $\boldsymbol{p}$ as a center and generating a $k \times k$ window around it. Then the $\boldsymbol{p}$ is resized to $\mathbb{R}^{T\times(k^2\times J)\times C_{in}}$. Now, the 2D poses $\boldsymbol{p}$ data is a 2D vector, which means the x and y coordinates of the human keypoints.

We denote $\bar{\boldsymbol{F}}^{l-1}$ and $\boldsymbol{X}^{l-1}$ as the last layer outputs of the APP module. As for the first layer of APP, the offsets $\boldsymbol{\Delta}^0$ are used to extract the initialized $\bar{\boldsymbol{F}}^0$ and $\boldsymbol{X}^0$:

$$\begin{aligned} \{\boldsymbol{H}_s\}_{s=1}^{S} &= \theta_f^0(\{\boldsymbol{H}_s\}_{s=1}^{S}), \\ \{\boldsymbol{F}_s\}_{s=1}^{S} &= GridSample(\{\boldsymbol{H}_s\}_{s=1}^{S}, \boldsymbol{\Delta}^0), \\ \bar{\boldsymbol{F}}^0 &= \frac{1}{S}\sum_{s=1}^{S} \boldsymbol{F}_s, \\ \boldsymbol{X}^0 &= AveragePool(\bar{\boldsymbol{F}}^0) \end{aligned} \tag{2}$$

Where θ_f is a group of 2D convolution operators to project multi-level feature maps into the same dimension d, the implementation of $GridSample$ is consistent with **nn.functional.grid_sample** in PyTorch, $AveragePool$ is to get the mean value of each $k\times k$ window.

Our proposed APP module consists of L stacked layers. In detail, the calculation process of each layer APP module is defined as follows:

$$\bar{\boldsymbol{F}}^l, \boldsymbol{X}^l = APP^l(\bar{\boldsymbol{F}}^{l-1}, \boldsymbol{X}^{l-1}, \boldsymbol{p}, \{H_i\}_{s=1}^{S}, \boldsymbol{P}) \tag{3}$$

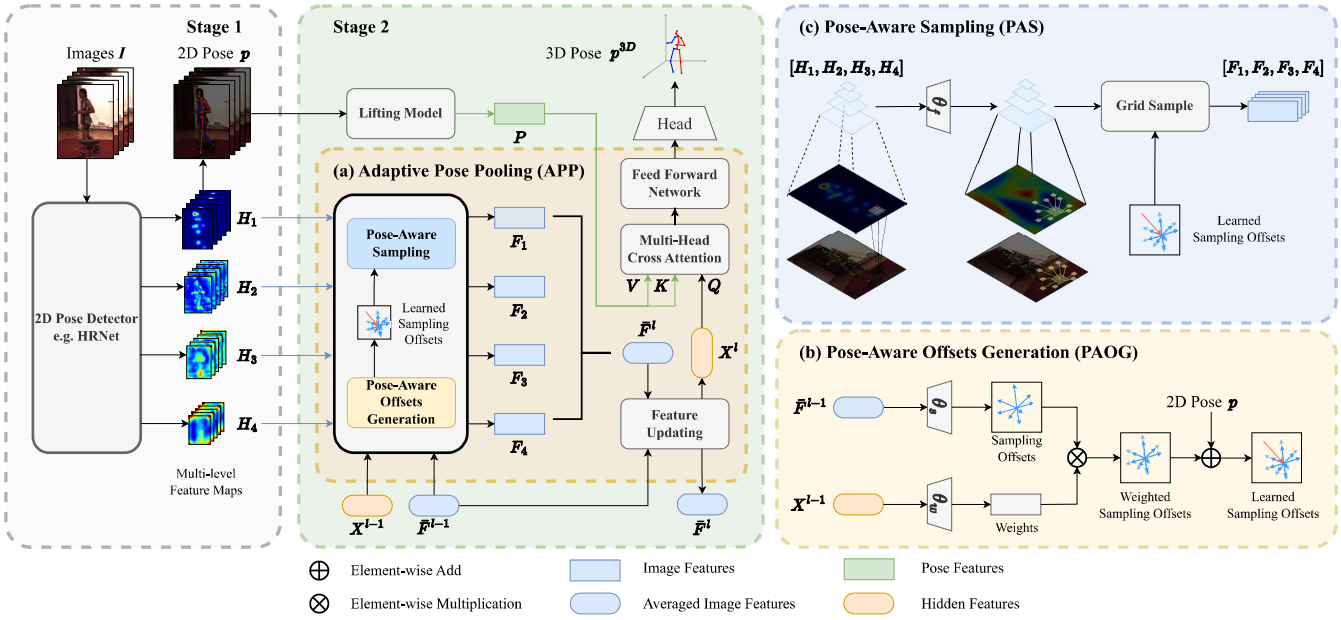


Figure 2: Our proposed APP module. (a) APP module contains three novel sub-modules: Pose-Aware Offsets Generation (PAOG), Pose-Aware Sampling (PAS), and Spatial Temporal Information Fusion (STIF). STIF comprises Feature Updating and Multi-Head Cross Attention, which is not shown in this figure. Feature Updating is used to obtain $\bar{F}^l$ and X^l. MHCA fuses the hidden features X^l with pose features P. (b) PAOG learns sampling offsets and utilizes these offsets to guide feature extraction. (c) PAS leverages learned offsets of PAOG and 2D human poses to sample multi-level feature maps.

In the last layer of the APP module, a simple regression head consisting of two fully connected layers estimates the final 3D human pose given by X^l.

3.2 Adaptive Pose Pooling

3.2.1 Pose-Aware Offsets Generation. Inspired by deformable convolution and deformable attention [9, 52], we propose the PAOG module as shown in Figure 2 (b). PAOG adaptively learns offset points during training. It takes the previous layer features $\bar{F}^{l-1}$ and X^{l-1}, and the 2D poses P_p^{2D} as input to generate learned offsets Δ^l, which is defined as follows:

$$\begin{aligned}\Delta_p^l &= PAOG^l(\bar{F}^{l-1}, X^{l-1}, p) \\ &= \theta_s^l(\bar{F}^{l-1}) \otimes \theta_w^l(X^{l-1}) + p\end{aligned} \tag{4}$$

Where $\otimes$ denotes element-wise multiplication, θ_s^l and θ_w^l generate offset points based on the image center and their corresponding weights. Note that they are both fully connected layers.

3.2.2 Pose-Aware Sampling. Using the offset points Δ_p^l generated by PAOG and the previous averaged image features $\bar{F}^{l-1}$, PAS also utilizes the multi-level feature maps $\{H\}_{s=1}^S \in \mathbb{R}^{t \times C_s \times H_s \times W_s}$ extracted from the 2D pose detector. We follow Deformable-DETR [52] before sampling features from the multi-level feature maps F. They are projected into the same dimension d as in Eq. (2). The sampling process is then defined as:

$$\begin{aligned}\{H_s\}_{s=1}^S &= \theta_f^l(\{H_s\}_{s=1}^S), \\ \{F_s\}_{s=1}^S &= GridSample(\{H_s\}_{s=1}^S, \Delta^l), \\ \bar{F}^l &= \frac{1}{S}\sum_{s=1}^{S} F_s\end{aligned} \tag{5}$$

3.2.3 Feature Updating. At the l layer of the APP module, we update the averaged image features $\bar{F}^l$ and hidden features X^l as follows:

$$\begin{aligned}X^l &= \theta_k^l(\bar{F}^l), \\ \bar{F}^l &= \alpha\bar{F}^{l-1} + (1-\alpha)\bar{F}^l\end{aligned} \tag{6}$$

Where $\theta_k^l(\cdot)$ represents a convolution operation with a kernel size of k^2, whose size is the same as the previously mentioned $k \times k$ window, this operation allows the PAS to learn the weights of k^2 features within the window to calculate X^l given by $\bar{F}^l$. α is a parameter controlling the magnitude of the update for $\bar{F}^l$.

3.2.4 Multi-Head Cross Attention. MHCA enables the APP module to harness the temporal features extracted by the multi-frame lifting models while interacting with the spatial features extracted by PAS. We first combine the pose feature P extracted with positional embeddings PE, initialized to zero, and updated during training.

We take $\boldsymbol{X}^l$ as $\boldsymbol{Q}$, $\boldsymbol{P}$ as $\boldsymbol{K}$, and $\boldsymbol{V}$, then we perform as follows:

$$\begin{aligned} \boldsymbol{Q} &= \boldsymbol{X}^l\boldsymbol{W}_{\boldsymbol{Q}}, \boldsymbol{K} = \boldsymbol{P}\boldsymbol{W}_{\boldsymbol{K}}, \boldsymbol{V} = \boldsymbol{P}\boldsymbol{W}_{\boldsymbol{V}}, \\ \boldsymbol{X}^l &= MHCA^l(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) \\ &= Concat(head_1, ..., head_h), \\ head_j &= Softmax\left(\frac{\boldsymbol{Q}_j\boldsymbol{K}_j}{\sqrt{d_P}}\right)\boldsymbol{V}_j \end{aligned} \tag{7}$$

Where $\boldsymbol{W}_{\boldsymbol{Q}} \in \mathbb{R}^{d\times d}$, $\boldsymbol{W}_{\boldsymbol{K}} \in \mathbb{R}^{d_P\times d}$, and $\boldsymbol{W}_{\boldsymbol{V}} \in \mathbb{R}^{d_P\times d}$, $j(j = 1, ..., h)$ represents the j-th attention head, and h is the number of attention heads.

3.2.5 Feed Forward Network. Like many monocular 3D HPE transformer-based methods, we apply an FFN at the end to finish computing the current layer. We use GeLU [15] as the activation function, as outlined in Eq. (8).

$$\boldsymbol{H}^l = FFN^l = FC(GeLU(FC(\boldsymbol{X}^l))) \tag{8}$$

3.2.6 Loss Function. The loss function of our proposed APP module is the same as [27, 51]. In addition to the widely used MPJPE loss, we also incorporate the Mean Per-Joint Velocity Error (MPJVE) loss and the normalized MPJPE [36] loss. Let $\mathcal{L}_p$, $\mathcal{L}_v$, and $\mathcal{L}_s$ denote these three losses. $\mathcal{L}_p$ calculates the L_2 distance between estimated 3D human poses $\hat{\boldsymbol{p}}$ and ground truth $\boldsymbol{g}$, while $\mathcal{L}_v$ minimizes the difference between $\Delta\hat{\boldsymbol{p}}$ and $\Delta\boldsymbol{g}$. $\mathcal{L}_p$ and $\mathcal{L}_v$ are thus given by:

$$\begin{aligned} \mathcal{L}_p &= \sum_{t=1}^{T}\sum_{j=1}^{J}\|\hat{\boldsymbol{p}}_{tj} - \boldsymbol{g}_{tj}\|_2, \\ \mathcal{L}_v &= \sum_{t=2}^{T}\sum_{j=1}^{J}\|\Delta\hat{\boldsymbol{p}}_{tj} - \Delta\boldsymbol{g}_{tj}\|_2 \end{aligned} \tag{9}$$

Where $\Delta\hat{\boldsymbol{p}}_t = \hat{\boldsymbol{p}}_t - \hat{\boldsymbol{p}}_{t-1}$, $\Delta\boldsymbol{g}_t = \boldsymbol{g}_t - \boldsymbol{g}_{t-1}$. For $\mathcal{L}_s$, estimated 3D poses $\hat{\boldsymbol{p}}$ is scaled according to ground truth $\boldsymbol{g}$. Then, we calculate MPJPE loss between scaled $\hat{\boldsymbol{p}}$ and $\boldsymbol{g}$. The total loss is formulated as follows:

$$\mathcal{L} = \gamma_p\mathcal{L}_p + \gamma_v\mathcal{L}_v + \gamma_s\mathcal{L}_s \tag{10}$$

where γ_p, γ_v, and γ_s are weights for $\mathcal{L}_p$, $\mathcal{L}_v$, and $\mathcal{L}_s$, respectively.

4 Experiments

4.1 Datasets and Evaluation Metrics

4.1.1 Human3.6M. Human3.6M [16] is a widely used large-scale 3D human pose estimation dataset. The dataset consists of images captured by four high-resolution cameras placed at the corners of a room to capture the human body from multiple angles. It includes 11 subjects performing 17 action scenarios (e.g., talking on the phone, taking photos), making it a benchmark dataset for 3D human pose estimation. We follow previous studies [27, 51] by using subjects 1, 5, 6, 7, and 8 for model training and subjects 9 and 11 for testing.

4.1.2 MPI-INF-3DHP. MPI-INF-3DHP [28] dataset comprises indoor and outdoor scenes. It contains over 1.3 million frames captured from 14 camera viewpoints, recording eight actions performed by eight subjects. The test set includes seven action classes and three different scenarios: green screen, non-green screen, and outdoor scenes.

4.1.3 Evaluation Metrics. For the Human3.6M dataset, we report the MPJPE and P-MPJPE. Before computing the error, the former aligns the estimated 3D pose with the ground truth root node (hip). At the same time, the latter requires rigid alignment based on the ground truth, including translation and rotation, before error calculation. For the MPI-INF-3DHP dataset, consistent with previous works [4, 27, 38, 47, 48], we report MPJPE, Percentage of Correct Keypoints (PCK) with a threshold of 150mm, and Area Under Curve (AUC).

4.2 Implementation Details

We adopt Transformer [42], MotionAGFormer-B [27], MixSTE [45], MotionBERT [51], and HoT [23] as the foundation models. These models' dimensions d_P are 128, 256, 512, and 128, respectively. **For a fair comparison, we use pretrained lifting models provided by the authors. Otherwise, we train them from the scratch.**

Although extensive results of the monocular 3D HPE models are illustrated in Table 1. However, single-frame and seq2frame models were excluded for lacking temporal information and facing fewer challenges than seq2seq models, respectively. **The reasons for choosing these five baselines are shown as follows.** (1) Transformer: Many seq2seq models are Transformer-based methods, so a spatial-temporal Transformer model is included. (2) MixSTE, (3) MotionBERT, (4) MotionAGFormer-B: SOTA methods in 2022, 2023, and 2024, respectively. (5) HoT: The recent method. The last column of Table B is the result of our proposed method.

YOLOv3 [35] is used to extract bounding boxes of humans in the images, and then we crop the images to 192×256 according to the bounding boxes. However, in CA-PF [47] experiments, the 3D ground truth human pose is directly used to extract bounding boxes, which is not feasible in real-world scenarios.

The lengths of 2D human pose sequences T on Human3.6M and MPI-INF-3DHP are set to 243 and 81, respectively, and images are uniformly sampled into sequence lengths $t = \frac{T}{9}$ for a tradeoff between performance and training cost. Additionally, t must be divisible by T. We extract feature maps from images using HRNet-w32 [39] with four resolutions. It is chosen as the pose detector, $C_s = 32^{s+1} \in \{32, 64, 128, 256\}, H_s = \frac{H}{4^{s+1}} \in \{\frac{H}{4}, \frac{H}{8}, \frac{H}{16}, \frac{H}{32}\}, W_s = \frac{W}{4^{s+1}} \in \{\frac{W}{4}, \frac{W}{8}, \frac{W}{16}, \frac{W}{32}\}$.

The parameters of our proposed APP module are mainly determined by the number of layers L, the number of attention heads h, the kernel size k, the dimension d of the APP module, and the dimension of hidden feature $\boldsymbol{F}_p$ used in the selected lifting model. Also, we employ DropPath [20] with the probability p. For Human3.6M and MPI-INF-3DHP, L, h, k, α, p, and d are set to 6, 8, 3, 0.9, 0.2, and 128, respectively. We train the APP module using the Adam optimizer [19]. 2D ground truth human pose is sent to the model for MPI-INF-3DHP. In Eq. (9), the weights of each part of the loss are set to 1.0, 20.0, and 0.5, respectively. All experiments are conducted on one NVIDIA RTX 3090 GPU.

4.3 Qualitative Comparison of Human3.6M

We conducted comparative evaluations with recent state-of-the-art (SOTA) models on the Human3.6M dataset. The results, summarized in Table 1, underscore the superiority of our proposed APP module when integrated with MotionBERT (finetune) and

Table 1: Qualitative comparisons on Human3.6M. The best and second best results are marked in red and blue respectively. (Same below) T: Number of the input frames. Seq2Seq: Estimate 3D human pose for every frame. †: Results reproduced in our experimental environment. ‡: Inference cost of D3DP[38] is much greater than training cost.

	Venue	Seq2Seq	T	2D Pose Detector	2D Pose	Param	MACs	MACs/frame	MPJPE↓	P-MPJPE↓
MHFormer [22]	CVPR'22		351	CPN	CPN	30.9M	7.0G	7096M	43.0	34.4
StridedTrans [21]	TMM'22		351	CPN	CPN	4.0M	0.8G	801M	43.7	35.2
MixSTE [45]	CVPR'22	✓	243	CPN	CPN	33.8M	147.6G	607M	40.9	32.6
P-STMO [37]	ECCV'22	✓	243	CPN	CPN	6.2M	0.7G	740M	42.8	34.4
CA-PF-CPN [47]	NeurIPS'23		1	CPN	CPN	14.1M	0.6G	610M	41.6	33.9
HDFormer [4]	IJCAI'23	✓	96	CPN	CPN	3.7M	0.6G	6M	42.6	33.1
PoseFormerV2 [48] (f=27)	CVPR'23		243	CPN	CPN	14.3M	0.5G	528M	45.2	35.6
STCFormer [41]	CVPR'23	✓	243	CPN	CPN	4.7M	19.6G	80M	41.0	32.0
STCFormer-L [41]	CVPR'23	✓	243	CPN	CPN	18.9M	78.2G	321M	40.5	31.8
UPS [11]	CVPR'23		243	CPN	CPN	-	-	-	40.8	32.5
D3DP‡ [38] (H=20,K=10, J-Agg)	ICCV'23	✓	243	CPN	CPN	34.6M	0.57G/228.2G	2M/939M	39.5	31.6
GLA-GCN [44]	ICCV'23		243	CPN	CPN	1.3M	1.5G	1556M	44.4	34.8
MotionAGFormer-B† [27]	WACV'24	✓	243	CPN	CPN	11.7M	64.8G	267M	**39.2**	**31.5**
KTPFormer [33]	CVPR'24	✓	243	CPN	CPN	33.7M	139.1G	572M	40.1	31.9
APP w. MixSTE (Ours)	-	✓	243	CPN	CPN	39.5M	174.9G	720M	39.8	32.1
APP w. MotionAGFormer-B (Ours)	-	✓	243	CPN	CPN	15.3M	90.1G	371M	**39.0**	**31.3**
Transformer† [42]	NeurIPS'17	✓	243	SH	SH	9.6M	46.2G	190M	39.2	32.8
MotionBERT [51] (scratch)	ICCV'23	✓	243	SH	SH	42.5M	185.6G	764M	39.2	32.9
MotionBERT [51] (finetune)	ICCV'23	✓	243	SH	SH	42.5M	185.6G	764M	**37.5**	-
MotionAGFormer-B [27]	WACV'24	✓	243	SH	SH	11.7M	64.8G	267M	38.4	32.6
MotionAGFormer-L [27]	WACV'24	✓	243	SH	SH	19.0M	78.3G	322M	38.4	**32.5**
HoT [23]	CVPR'24	✓	243	SH	SH	16.4M	33.5G	138M	39.8	33.5
HoT [23]	CVPR'24		243	SH	SH	16.0M	-	-	39.0	-
APP w. Transformer (Ours)	-	✓	243	CPN	SH	13.2M	70.7G	291M	39.0	32.7
APP w. HoT (Ours)	-	✓	243	CPN	SH	20.7M	50.8G	209M	39.7	33.4
APP w. MotionBERT (finetune) (Ours)	-	✓	243	CPN	SH	48.2M	211.8G	871M	**37.0**	**31.5**
MixSTE [45]	CVPR'22	✓	243	HRNet	HRNet	33.8M	147.6G	607M	39.8	**30.6**
CA-PF-HRNet-32 [47]	NeurIPS'23		1	HRNet-w32	CPN	14.1M	0.6G	610M	41.4	33.5
CA-PF-HRNet-48 [47]	NeurIPS'23		1	HRNet-w48	CPN	14.1M	0.6G	610M	39.8	32.7
HDFormer [4]	IJCAI'23	✓	96	HRNet	HRNet	3.7M	0.6G	6M	40.3	31.7
MotionAGFormer-B† [27]	WACV'24	✓	243	HRNet	HRNet	11.7M	64.8G	267M	**36.6**	**29.5**
APP w. Transformer (Ours)	-	✓	243	HRNet-w32	SH	13.2M	70.7G	291M	39.0	32.7
APP w. MixSTE (Ours)	-	✓	243	HRNet-w32	CPN	39.5M	174.9G	720M	39.9	32.1
APP w. MotionBERT (finetune) (Ours)	-	✓	243	HRNet-w32	SH	48.2M	211.8G	871M	37.0	31.6
APP w. HoT (Ours)	-	✓	243	HRNet-w32	SH	20.7M	50.8G	209M	39.7	33.4
APP w. MotionAGFormer-B (Ours)	-	✓	243	HRNet-w32	CPN	15.3M	90.1G	371M	38.9	31.3
APP w. MotionAGFormer-B (Ours)	-	✓	243	HRNet-w32	HRNet	15.3M	90.1G	371M	**36.2**	**29.5**

MotionAGFormer-B, attaining SOTA performance. Specifically, in comparison with lifting models MixSTE [45] and HDFormer [4], which also utilize HRNet-detected 2D human pose as input, our approach achieves notable improvements of 3.6mm (from 39.8mm to 36.2mm) and 4.1mm (from 40.3mm to 36.2mm) in MPJPE, and 1.1mm (from 30.6mm to 29.5mm) and 2.2mm (from 31.7mm to 29.5mm) in terms of P-MPJPE, respectively. What's more, our proposed method surpasses CA-PF-HRNet-48 [47] 3.6mm (from 39.8mm to 36.2mm) in MPJPE, and 3.2mm (from 32.7mm to 29.5mm).

Generally, CPN performs worse than SH and HRNet, and we have considered this. We point out the 2D pose and 2D pose detector of each model in Table 1. **Across all experiments, our method achieves SOTA with CPN, SH, and HRNet as 2D pose detectors.**

In Table 2, HRNet-w32[39] is adopted as the 2D pose detector. Our method exhibits better results in terms of MPJPE, with 1.6mm improvement compared to MotionBERT (finetune) [51] when ground truth 2D human pose is employed as input, it surpasses MotionAGFormer-L by 2.0mm. Additionally, we present another result, APP w. MotionAGFormer-B, which surpasses MotionAGFormer-B.

These two fair comparisons demonstrate that our proposed APP method enhances the performance of existing methods, including both MotionBERT (finetune) and MotionAGFormer-B.

In summary, these findings collectively attest to the efficacy and competitiveness of our proposed approach in advancing the state-of-the-art in monocular 3D human pose estimation.

4.4 Qualitative Comparison of MPI-INF-3DHP

In evaluating the MPI-INF-3DHP dataset, we aligned the sequence length T to 81 frames to maintain consistency with the experimental setup of MotionAGFormer-B [27]. As detailed in Table 2, our method demonstrates incremental enhancements across various evaluation metrics. Specifically, we observe marginal improvements in the AUC and MPJPE upon KTPFormer [33], with gains of 3.6 (from 85.9

Table 2: Qualitative comparisons using 2D Ground Truth on Human3.6M.

GT	Seq2Seq	T	MPJPE↓
MHFormer [22] CVPR'22		351	30.5
StridedTrans [21] TMM'22		351	28.5
MixSTE [45] CVPR'22	✓	81	25.9
MixSTE [45] CVPR'22	✓	243	21.6
P-STMO [37] ECCV'22	✓	243	29.3
D3DP [38] (H=20,K=10, J-Agg) ICCV'23	✓	243	19.6
GLA-GCN [44] ICCV'23		243	21.0
STCFormer [41] CVPR'23	✓	81	25.0
STCFormer-L [41] CVPR'23	✓	243	21.3
HDFormer [4] IJCAI'23	✓	96	21.6
MotionBERT [51] (scratch) ICCV'23	✓	243	17.8
MotionBERT [51] (finetune) ICCV'23	✓	243	**16.9**
KTPFormer [33] CVPR'24	✓	243	18.1
MotionAGFormer-B [27] WACV'24	✓	243	19.4
MotionAGFormer-L [27] WACV'24	✓	243	17.3
APP w. MotionAGFormer-B (Ours)	✓	243	19.1
APP w. MotionBERT (Ours)	✓	243	**15.3**

Table 3: Qualitative comparisons on MPI-INF-3DHP.

Method	T	AUC↑	MPJPE↓	PCK↑
D3DP [38] (H=20,K=10, J-Agg) ICCV'23	243	78.2	29.7	97.7
HDFormer [4] IJCAI'23	32	64.0	51.5	96.8
HDFormer [4] IJCAI'23	96	72.9	37.2	**98.7**
PoseFormerV2 [48] CVPR'23	81	78.8	27.8	97.9
CA-PF-HRNet-32 [47] NeurIPS'23	1	75.4	32.7	98.0
CA-PF-HRNet-48 [47] NeurIPS'23	1	76.3	31.4	98.2
MotionAGFormer-B [27] WACV'24	81	84.2	18.2	98.3
MotionAGFormer-L [27] WACV'24	81	85.3	**16.2**	98.2
KTPFormer [33] CVPR'24	27	84.4	19.2	**98.9**
KTPFormer [33] CVPR'24	81	**85.9**	16.7	**98.9**
APP w. MotionAGFormer-B (Ours)	81	**89.5**	**12.7**	**98.9**

to 89.5) and 4mm (from 16.7mm to 12.7mm), respectively. Moreover, the PCK of our method is the same as KTPFormer [33] because the threshold of PCK is set to 150mm. When comparing our method with the baseline model MotionAGFormer-B[27], there are huge improvements for AUC (from 84.2 to 89.5) and MPJPE (from 18.2mm to 12.7mm). These findings underscore the efficacy and robustness of our proposed method in enhancing monocular 3D human pose estimation performance on the MPI-INF-3DHP dataset.

4.5 Ablation Study

All ablation studies are conducted on the Human3.6M dataset, with MotionAGFormer-B [27] chosen as the baseline model and 2D pose detected by HRNet [39]. Initially, we conducted experiments on parameter selections for the APP module. Table 4 indicates that our proposed method achieves the best performance when L, k, α, p, and d are selected as 6, 3, 0.9, 0.2, and 128, respectively.

Then, we conducted ablation experiments on the submodules of our proposed APP module. In Table 5, the first row presents partial components of three submodules: MHCA, PAOG, and PAS. MHCA:

Table 4: Ablation study of the parameters on Human3.6M.

L	k	α	p	d	Params	MPJPE↓	P-MPJPE↓
6	3	0.9	0.2	128	3.58M	**36.2**	**29.5**
4	3	0.9	0.2	256	7.50M	36.5	29.7
4	3	0.9	0.4	128	2.76M	**36.3**	**29.6**
6	5	0.9	0.2	128	5.15M	36.5	29.7
6	3	0.6	0.2	128	3.58M	36.6	30.0
6	3	0.9	0	128	3.58M	36.5	29.9

Use MHCA, otherwise an identity layer is adopted without using any pose features $\boldsymbol{P}$. Weights: Apply weights to the learned offsets. PoseAware: Choose the detected 2D pose as the pivot; otherwise, the center of the image will be selected. Conv: Convolution is performed on the sampled features; otherwise, the mean operation is carried out directly.

Table 5: Ablation study of the components on Human3.6M.

	MHCA	Weights	PoseAware	Conv	Params	MPJPE↓	P-MPJPE↓
1	✓				2.69M	36.6	29.8
2		✓			2.30M	62.0	43.1
3			✓		2.30M	68.0	44.7
4				✓	3.18M	67.2	45.7
5	✓	✓			2.69M	**36.2**	**29.6**
6	✓		✓		2.69M	36.7	29.9
7	✓			✓	3.58M	36.4	**29.6**
8		✓	✓		2.30M	63.6	43.2
9		✓		✓	3.18M	59.1	40.0
10			✓	✓	3.18M	71.0	45.1
11	✓	✓	✓		2.69M	36.5	30.0
12	✓	✓		✓	3.58M	**36.3**	29.9
13	✓		✓	✓	3.58M	36.4	**29.6**
14		✓	✓	✓	3.18M	57.0	39.1
15	✓	✓	✓	✓	3.58M	**36.2**	**29.5**

MHCA has the most significant impact on the APP model's performance. We show different QKV settings of MHCA in Table 6. However, the other three components also enhance the model's performance to varying degrees.

Table 6: Ablaton study of the different QKV settings in MHCA.

	Q	K	V	MPJPE↓	P-MPJPE↓
1	X	P	P	**36.2**	**29.5**
2	X	X	P	**36.3**	**29.7**
3	X	P	X	36.4	30.0
4	P	X	X	**36.3**	**29.7**
5	P	P	X	**36.3**	29.8
6	P	X	P	**36.3**	29.8

Finally, we conducted ablaton experiments on image frames t setting. Although the MACs (24.83G) of the APP module are not affected by the number of frames, yet image frames t need more computation in training stage because we need multi-level featuremaps extracted by 2D pose detector. Furthermore, when $t \geq 81$, the training time becomes very long. Therefore, $t = 27 = \frac{T}{9}$ is tradeoff between accuracy and efficiency. In inference stage, this problem no longer appears, owing to that 2D pose detector is used to estimate 2D pose for every frame.

Table 7: Ablation study of the image frames t on Human3.6M.

t	Params	MPJPE↓	P-MPJPE↓
1	3.58M	36.5	30.0
3	3.58M	36.5	**29.8**
9	3.18M	**36.3**	**29.6**
27	3.58M	**36.2**	**29.6**

4.6 Visualization

To validate the generalization ability of the model, we selected the MotionBERT (finetune) [51] and MotionAGFormer-B [27] as foundation models, which were trained on 2D poses extracted by SH [30] on Human3.6M dataset, for comparison with our proposed APP module. For the visual analysis in Figures 3 and 4, MotionAGFormer-B was selected as the foundation model.

4.6.1 Analysis of MHCA Module. We visualized the attention maps of the MHCA submodule at each layer of our proposed APP module. The values of each attention map were normalized to $[0, 1]$. Each attention map extracted at every layer is a 17×17 matrix. Figure 3 shows that although each layer has different focuses, the APP module can capture the relationships between key points.

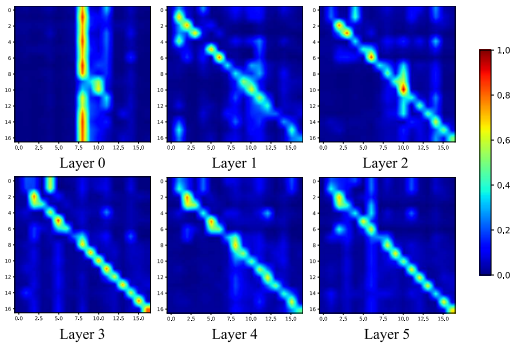


Figure 3: Visualization of the attention maps of each layer of MHCA in our proposed APP module.

4.6.2 Analysis of PAOG Module. Figure 4 illustrates the sampling points learned by our proposed PAOG submodule. The first row in the figure is appended for observation, while the second to fifth rows represent feature maps extracted at different resolutions. Each layer of the PAOG submodule can adaptively learn sampling points based on the distribution of feature maps at different resolutions. The model tends to use features near the human body.

4.6.3 Analysis of Human3.6M and In-the-wild Videos. A challenging sample of Human3.6M testset with the ground truth is shown in 5. The second row is the visualization of video from the internet. It can be seen that APP w. MotionBERT (finetune) reaches better results than MotionBERT (finetune).

5 Conclusion

In this paper, we introduce a flexible module called APP crafted to mesh with various multi-frame lifting models. This versatile module utilizes the hidden features extracted by lifting models with the feature maps obtained from the 2D pose detector. APP excels in simultaneously extracting spatial and temporal features through this collaborative approach module. Extensive experiments highlight the reliability of our module, showcasing its ability to enhance the performance metrics of lifting models while consistently achieving state-of-the-art results across diverse scenarios. Notably, the APP module presents remarkable adaptability, maintaining its effectiveness even when paired with different settings. A noteworthy feature is its capacity to enhance performance without requiring structural changes to the existing lifting model architecture.

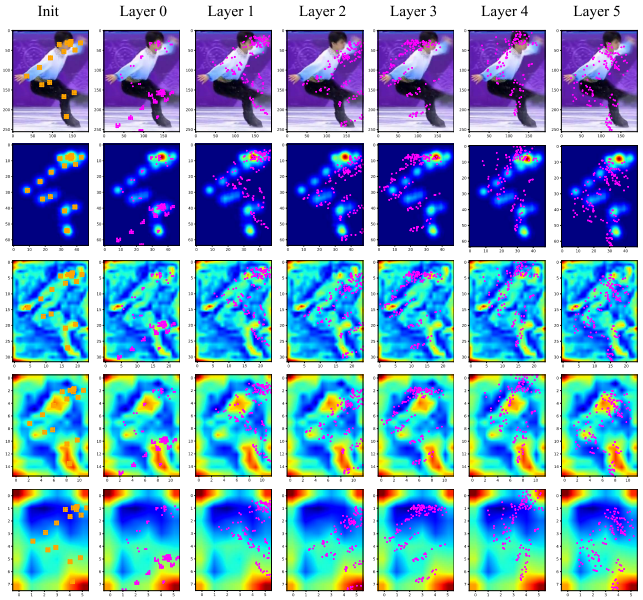


Figure 4: Visualization of the initialized/learned sampling points learned by each layer, where the initialized/learned sampling points are marked in orange and magenta, respectively. Each sampling point is represented by a cross (x).

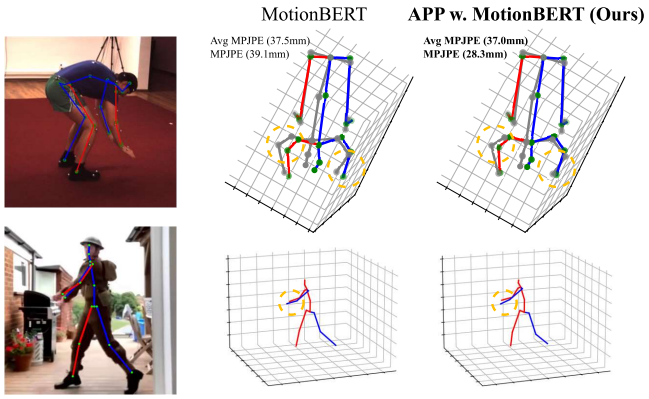


Figure 5: Gray pose is the 3D ground truth on Human3.6M dataset. Two examples come from Human3.6M dataset (Row 1) and an in-the-wild video (Row 2), respectively. We highlight the differences with orange circles.

Acknowledgments

This work was supported by Shenzhen Innovation in Science and Technology Foundation for The Excellent Youth Scholars (No. RCYX20231211090248064), National Natural Science Foundation of China (No. 62203476).

References

[1] Peter Bauer, Arij Bouazizi, Ulrich Kressel, and Fabian B Flohr. 2023. Weakly supervised multi-modal 3d human body pose estimation for autonomous driving. In *2023 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 1–7.

[2] Tobias Baumgartner and Stefanie Klatt. 2023. Monocular 3D human pose estimation for sports broadcasts using partial sports field registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. 5108–5117.

[3] Yujun Cai, Liuhao Ge, Jun Liu, Jianfei Cai, Tat-Jen Cham, Junsong Yuan, and Nadia Magnenat Thalmann. 2019. Exploiting spatial-temporal relationships for 3D pose estimation via graph convolutional networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2272–2281.

[4] Hanyuan Chen, Jun-Yan He, Wangmeng Xiang, Zhi-Qi Cheng, Wei Liu, Hanbing Liu, Bin Luo, Yifeng Geng, and Xuansong Xie. 2023. HDFormer: high-order directed transformer for 3D human pose estimation. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*. 581–589.

[5] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. 2018. Cascaded pyramid network for multi-person pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7103–7112.

[6] Sungho Chun, Sungbum Park, and Ju Yong Chang. 2023. Learnable human mesh triangulation for 3D human pose and shape estimation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2850–2859.

[7] Hai Ci, Chunyu Wang, Xiaoxuan Ma, and Yizhou Wang. 2019. Optimizing network structure for 3d human pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2262–2271.

[8] Huseyin Coskun, David Joseph Tan, Sailesh Conjeti, Nassir Navab, and Federico Tombari. 2018. Human motion analysis with deep metric learning. In *Proceedings of the European conference on computer vision (ECCV)*. 667–683.

[9] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. 2017. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*. 764–773.

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICLR*.

[11] Lin Geng Foo, Tianjiao Li, Hossein Rahmani, Qiuhong Ke, and Jun Liu. 2023. Unified pose sequence modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 13019–13030.

[12] Jia Gong, Zhipeng Fan, Qiuhong Ke, Hossein Rahmani, and Jun Liu. 2022. Meta agent teaming active learning for pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 11079–11089.

[13] Renshu Gu, Gaoang Wang, and Jenq-Neng Hwang. 2019. Efficient multi-person hierarchical 3d pose estimation for autonomous driving. In *2019 IEEE conference on multimedia information processing and retrieval (MIPR)*. IEEE, 163–168.

[14] Yihui He, Rui Yan, Katerina Fragkiadaki, and Shoou-I Yu. 2020. Epipolar transformers. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*. 7779–7788.

[15] Dan Hendrycks and Kevin Gimpel. 2016. Bridging Nonlinearities and Stochastic Regularizers with Gaussian Error Linear Units. (2016).

[16] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. 2013. Human3.6M: Large scale datasets and predictive methods for 3D human sensing in natural environments. *IEEE TPAMI* 36, 7 (2013), 1325–1339.

[17] Karim Iskakov, Egor Burkov, Victor Lempitsky, and Yury Malkov. 2019. Learnable triangulation of human pose. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 7718–7727.

[18] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. 2018. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7122–7131.

[19] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[20] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. 2016. FractalNet: Ultra-Deep Neural Networks without Residuals. In *International Conference on Learning Representations*.

[21] Wenhao Li, Hong Liu, Runwei Ding, Mengyuan Liu, Pichao Wang, and Wenming Yang. 2022. Exploiting Temporal Contexts with Strided Transformer for 3D Human Pose Estimation. *IEEE TMM* 25 (2022), 1282–1293.

[22] Wenhao Li, Hong Liu, Hao Tang, Pichao Wang, and Luc Van Gool. 2022. MHFormer: Multi-Hypothesis Transformer for 3D Human Pose Estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 13147–13156.

[23] Wenhao Li, Mengyuan Liu, Hong Liu, Pichao Wang, Jialun Cai, and Nicu Sebe. 2024. Hourglass Tokenizer for Efficient Transformer-Based 3D Human Pose Estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[24] Huei-Yung Lin and Ting-Wen Chen. 2010. Augmented reality with human body interaction based on monocular 3d pose estimation. In *International Conference on Advanced Concepts for Intelligent Vision Systems*. Springer, 321–331.

[25] Ruixu Liu, Ju Shen, He Wang, Chen Chen, Sen-ching Cheung, and Vijayan Asari. 2020. Attention mechanism exploits temporal contexts: Real-time 3D human pose reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5064–5073.

[26] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. 2017. A simple yet effective baseline for 3D human pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2640–2649.

[27] Soroush Mehraban, Vida Adeli, and Babak Taati. 2024. MotionAGFormer: Enhancing 3D Human Pose Estimation with a Transformer-GCNFormer Network. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 6920–6930.

[28] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. 2017. Monocular 3D human pose estimation in the wild using improved CNN supervision. In *3DV*. 506–516.

[29] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. 2017. Vnect: Real-time 3d human pose estimation with a single rgb camera. *Acm transactions on graphics (tog)* 36, 4 (2017), 1–14.

[30] Alejandro Newell, Kaiyu Yang, and Jia Deng. 2016. Stacked hourglass networks for human pose estimation. In *Proceedings of the European conference on computer vision (ECCV)*. 483–499.

[31] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis. 2017. Coarse-to-fine volumetric prediction for single-image 3D human pose. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7025–7034.

[32] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 2019. 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*. 7753–7762.

[33] Jihua Peng, Yanghong Zhou, and PY Mok. 2024. KTPFormer: Kinematics and Trajectory Prior Knowledge-Enhanced Transformer for 3D Human Pose Estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[34] N Dinesh Reddy, Laurent Guigues, Leonid Pishchulin, Jayan Eledath, and Srinivasa G Narasimhan. 2021. Tessetrack: End-to-end learnable multi-person articulated 3d pose tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 15190–15200.

[35] Joseph Redmon and Ali Farhadi. 2018. YOLOv3: An Incremental Improvement. *arXiv* (2018).

[36] Helge Rhodin, Mathieu Salzmann, and Pascal Fua. 2018. Unsupervised geometry-aware representation for 3d human pose estimation. In *Proceedings of the European conference on computer vision (ECCV)*. 750–767.

[37] Wenkang Shan, Zhenhua Liu, Xinfeng Zhang, Shanshe Wang, Siwei Ma, and Wen Gao. 2022. P-STMO: Pre-Trained Spatial Temporal Many-to-One Model for 3D Human Pose Estimation. In *Proceedings of the European conference on computer vision (ECCV)*.

[38] Wenkang Shan, Zhenhua Liu, Xinfeng Zhang, Zhao Wang, Kai Han, Shanshe Wang, Siwei Ma, and Wen Gao. 2023. Diffusion-based 3d human pose estimation with multi-hypothesis aggregation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 14761–14771.

[39] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. 2019. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 5693–5703.

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

[41] Zhenhua Tang, Zhaofan Qiu, Yanbin Hao, Richang Hong, and Ting Yao. 2023. 3D Human Pose Estimation With Spatio-Temporal Criss-Cross Attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4790–4799.

[42] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *NeurIPS*. 5998–6008.

[43] Jingwei Xu, Zhenbo Yu, Bingbing Ni, Jiancheng Yang, Xiaokang Yang, and Wenjun Zhang. 2020. Deep kinematics analysis for monocular 3d human pose estimation. In *Proceedings of the IEEE/CVF Conference on computer vision and Pattern recognition*. 899–908.

[44] Bruce XB Yu, Zhi Zhang, Yongxu Liu, Sheng-hua Zhong, Yan Liu, and Chang Wen Chen. 2023. Gla-gcn: Global-local adaptive graph convolutional network for 3d human pose estimation from monocular video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 8818–8829.

[45] Jinlu Zhang, Zhigang Tu, Jianyu Yang, Yujin Chen, and Junsong Yuan. 2022. MixSTE: Seq2seq Mixed Spatio-Temporal Encoder for 3D Human Pose Estimation in Video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 13232–13242.

[46] Zhe Zhang, Chunyu Wang, Weichao Qiu, Wenhu Qin, and Wenjun Zeng. 2021. Adafuse: Adaptive multiview fusion for accurate human pose estimation in the wild. *International Journal of Computer Vision* 129 (2021), 703–718.

[47] Qitao Zhao, Ce Zheng, Mengyuan Liu, and Chen Chen. 2023. A Single 2D Pose with Context is Worth Hundreds for 3D Human Pose Estimation. In *Thirty-seventh Conference on Neural Information Processing Systems*.

[48] Qitao Zhao, Ce Zheng, Mengyuan Liu, Pichao Wang, and Chen Chen. 2023. PoseFormerV2: Exploring Frequency Domain for Efficient and Robust 3D Human Pose Estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8877–8886.

[49] Ce Zheng, Sijie Zhu, Matias Mendieta, Taojiannan Yang, Chen Chen, and Zhengming Ding. 2021. 3D Human Pose Estimation with Spatial and Temporal Transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 11656–11665.

[50] Kun Zhou, Xiaoguang Han, Nianjuan Jiang, Kui Jia, and Jiangbo Lu. 2019. Hemlets pose: Learning part-centric heatmap triplets for accurate 3d human pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2344–2353.

[51] Wentao Zhu, Xiaoxuan Ma, Zhaoyang Liu, Libin Liu, Wayne Wu, and Yizhou Wang. 2023. MotionBERT: A Unified Perspective on Learning Human Motion Representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 15085–15099.

[52] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. 2020. Deformable DETR: Deformable Transformers for End-to-End Object Detection. In *International Conference on Learning Representations*.