MotionBERT: A Unified Perspective on Learning Human Motion Representations

Wentao Zhu[1] Xiaoxuan Ma[1] Zhaoyang Liu[2] Libin Liu[1] Wayne Wu[2] Yizhou Wang[1†]

[1] Peking University [2] Shanghai AI Laboratory

{wtzhu,maxiaoxuan,libin.liu,yizhou.wang}@pku.edu.cn

{zyliumy,wuwenyan0503}@gmail.com

Abstract

We present a unified perspective on tackling various human-centric video tasks by learning human motion representations from large-scale and heterogeneous data resources. Specifically, we propose a pretraining stage in which a motion encoder is trained to recover the underlying 3D motion from noisy partial 2D observations. The motion representations acquired in this way incorporate geometric, kinematic, and physical knowledge about human motion, which can be easily transferred to multiple downstream tasks. We implement the motion encoder with a Dual-stream Spatio-temporal Transformer (DSTformer) neural network. It could capture long-range spatio-temporal relationships among the skeletal joints comprehensively and adaptively, exemplified by the lowest 3D pose estimation error so far when trained from scratch. Furthermore, our proposed framework achieves state-of-the-art performance on all three downstream tasks by simply finetuning the pretrained motion encoder with a simple regression head (1-2 layers), which demonstrates the versatility of the learned motion representations. Code and models are available at https://motionbert.github.io/

1. Introduction

Perceiving and understanding human activities have long been a core pursuit of machine intelligence. To this end, researchers define various tasks to estimate *human-centric* semantic labels from videos, *e.g*. skeleton keypoints [13, 33], action classes [60, 116], and surface meshes [42, 66]. While significant progress has been made in each of these tasks, they tend to be modeled in isolation, rather than as interconnected problems. For example, Spatial Temporal Graph Convolutional Networks (ST-GCN) have been applied to modeling spatio-temporal relationship of human joints in both 3D pose estimation [12, 109] and action recognition [89, 116], but their connections have not been fully explored. Intuitively, these models should all have learned to identify typical human motion patterns, despite being designed for different problems. Nonetheless, current methods fail to mine and utilize such commonalities across the tasks. Ideally, we could develop a unified *human-centric* video representation that can be shared across all relevant tasks.

[†]Yizhou Wang is with Center on Frontiers of Computing Studies, School of Computer Science, Peking University and Institute for Artificial Intelligence, Peking University.

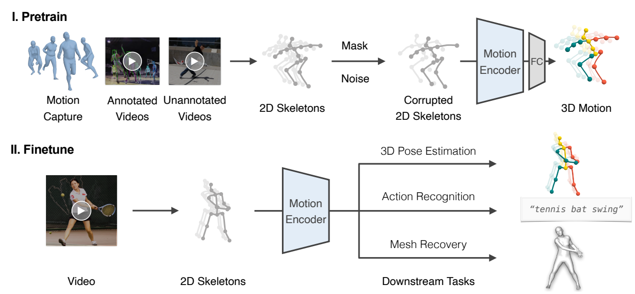

Figure 1. **Framework overview.** We utilize a motion encoder to learn human motion representations via recovering 3D human motion from corrupted 2D skeleton sequences. To adapt to different downstream tasks, we finetune the pretrained motion representations with a linear layer or a simple MLP.

One significant challenge to developing such a representation is the heterogeneity of available data resources. Motion capture (Mocap) systems [36, 71] provide high-fidelity 3D motion data obtained with markers and sensors, but the appearances of captured videos are usually constrained to simple indoor scenes. Action recognition datasets provide annotations of the action semantics, but they either contain no human pose labels [15, 88] or feature limited motion of daily activities [59, 60, 86]. In contrast, in-the-wild human videos offer a vast and diverse range of appearance and motion. However, obtaining precise 2D pose annotations requires considerable effort [3], and acquiring ground-truth (GT) 3D joint locations is almost impossible. Consequently, most existing studies focus on a specific task using a single type of human motion data, and they are not able to enjoy the advantages of other data resources.

In this work, we provide a new perspective on learning human motion representations. The key idea is that we can learn a versatile human motion representation from heterogeneous data resources in a *unified* manner, and utilize the representation to handle different downstream tasks in a *unified* way. We present a two-stage framework, consisting of pretraining and finetuning, as depicted in Figure 1. In the pretraining stage, we extract 2D skeleton sequences from diverse motion data sources and corrupt them with random masks and noises. Subsequently, we train the motion encoder to recover the 3D motion from the corrupted 2D skeletons. This challenging pretext task intrinsically requires the motion encoder to i) infer the underlying 3D human structures from its temporal movements; ii) recover the erroneous and missing observations. In this way, the motion encoder implicitly captures human motion commonsense such as joint linkages, anatomical constraints, and temporal dynamics. In practice, we propose *Dual-stream Spatio-temporal Transformer (DSTformer)* as the motion encoder to capture the long-range relationship among skeleton keypoints. We suppose that the motion representations learned from large-scale and diversified data resources could be shared across different downstream tasks and benefit their performance. Therefore, for each downstream task, we adapt the pretrained motion representations using task-specific training data and supervisory signals with a simple regression head.

In summary, the contributions of this work are three-fold: 1) We provide a new perspective on solving various human-centric video tasks through a shared framework of learning human motion representations. 2) We propose a pretraining method to leverage the large-scale yet heterogeneous human motion resources and learn generalizable human motion representations. Our approach could take advantage of the precision of 3D mocap data and the diversity of in-the-wild RGB videos at the same time. 3) We design a dual-stream Transformer network with cascaded spatio-temporal self-attention blocks that could serve as a general backbone for human motion modeling. The experiments demonstrate that the above designs enable a versatile human motion representation that can be transferred to multiple downstream tasks, outperforming the task-specific state-of-the-art methods.

2. Related Work

Learning Human Motion Representations. Early works formulate human motion with Hidden Markov Models [49, 101] and graphical models [47, 92]. Kanazawa *et al*. [39] design a temporal encoder and a hallucinator to learn representations of 3D human dynamics. Zhang *et al*. [124] predict future 3D dynamics in a self-supervised manner. Sun *et al*. [95] further incorporate action labels with an action memory bank. From the action recognition perspective, a variety of pretext tasks are designed to learn motion representations in a self-supervised manner, including future prediction [93], jigsaw puzzle [56], skeleton-contrastive [100], speed change [94], cross-view consistency [58], and contrast-reconstruction [110]. Similar techniques are also explored in tasks like motion assessment [31, 80] and motion retargeting [119, 130]. These methods leverage homogeneous motion data, design corresponding pretext tasks, and apply them to a specific downstream task. In this work, we propose a unified pretrain-finetune framework to incorporate heterogeneous data resources and demonstrate its versatility in various downstream tasks.

3D Human Pose Estimation. Recovering 3D human poses from monocular RGB videos is a classical problem, and the methods can be categorized into two categories. The first is to estimate 3D poses with CNN directly from images [77, 97, 128]. However, one limitation of these approaches is that there is a trade-off between 3D pose precision and appearance diversity due to current data collection techniques. The second category is to extract the 2D pose first, then lift the estimated 2D pose to 3D with a separate neural network. The lifting can be achieved via Fully Connected Network [28, 73], Temporal Convolutional Network (TCN) [21, 83], GCN [12, 27, 109], and Transformer [52, 87, 126, 127]. Our framework is built upon the second category as we use the proposed DSTformer to accomplish 2D-to-3D lifting.

Skeleton-based Action Recognition. The pioneering works [69, 108, 120] point out the inherent connection between action recognition and human pose estimation. Towards modeling the spatio-temporal relationship among human joints, previous studies mainly employ LSTM [91, 129] and GCN [20, 51, 64, 89, 116]. Most recently, PoseConv3D [30] proposes to apply 3D-CNN on the stacked 2D joint heatmaps and achieves improved results. In addition to the fully-supervised action recognition task, NTU-RGB+D-120 [60] brings attention to the challenging one-shot action recognition problem. To this end, SL-DML [76] applies deep metric learning to multi-modal signals. Sabater *et al*. [85] explores one-shot recognition in therapy scenarios with TCN. We demonstrate that the pretrained motion representations could generalize well to action recognition tasks, and the pretrain-finetune framework is a suitable solution for the one-shot challenges.

Human Mesh Recovery. Based on the parametric human models such as SMPL [66], many research works [38, 70, 78, 115, 125] focus on regressing the human mesh from a single image. SPIN [44] additionally incorporates fitting the body

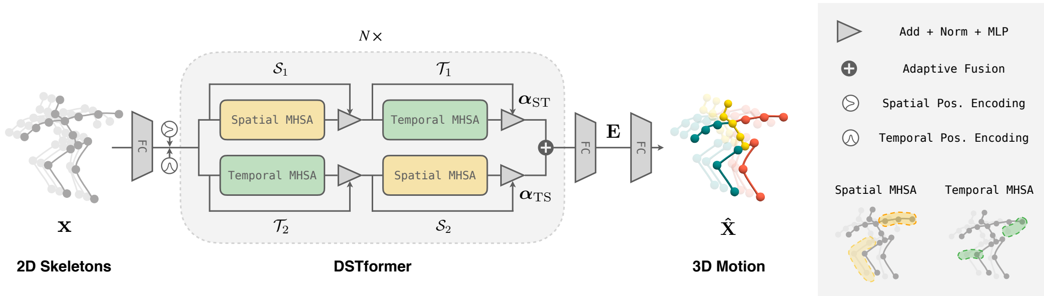

Figure 2. **Model architecture.** We propose the Dual-stream Spatio-temporal Transformer (DSTformer) as a general backbone for human motion modeling. DSTformer consists of N dual-stream-fusion modules. Each module contains two branches of spatial or temporal MHSA and MLP. The Spatial MHSA models the connection among different joints within a timestep, while the Temporal MHSA models the movement of one joint.

model to 2D joints in the training loop. Despite their promising per-frame results, these methods yield jittery and unstable results [42,123] when applied to videos. To improve their temporal coherence, PoseBERT [7] and SmoothNet [123] propose to employ a denoising and smoothing module to the single-frame predictions. Several works [23,39,42,99] take video clips as input to exploit the temporal cues. Another common problem is that paired images and GT meshes are mostly captured in constrained scenarios, which limits the generalization ability of the above methods. To that end, Pose2Mesh [24] proposes to first extract 2D skeletons using an off-the-shelf pose estimator, then lift them to 3D mesh vertices. Our approach is complementary to state-of-the-art human mesh recovery methods and could further improve their temporal coherence with the pretrained motion representations.

3. Method

3.1. Overview

As discussed in Section 1, our approach consists of two stages, namely unified pretraining and task-specific finetuning. In the first stage, we train a motion encoder to accomplish the 2D-to-3D lifting task, where we use the proposed DSTformer as the backbone. In the second stage, we finetune the pretrained motion encoder and a few new layers on the downstream tasks. We use 2D skeleton sequences as input for both pretraining and finetuning because they could be reliably extracted from all kinds of motion sources [3,9,71,81,96], and is more robust to variations [18,30]. Existing studies have shown the effectiveness of using 2D skeleton sequences for different downstream tasks [24,30,83,102]. We will first introduce the architecture of DSTformer, and then describe the training scheme in detail.

3.2. Network Architecture

Figure 2 shows the network architecture for 2D-to-3D lifting. Given an input 2D skeleton sequence $\mathbf{x} \in \mathbb{R}^{T\times J\times C_{\text{in}}}$, we first project it to a high-dimensional feature $\mathbf{F}^0 \in \mathbb{R}^{T\times J\times C_{\text{f}}}$, then add learnable spatial positional encoding $\mathbf{P}^{\text{S}}_{\text{pos}} \in \mathbb{R}^{1\times J\times C_{\text{f}}}$ and temporal positional encoding $\mathbf{P}^{\text{T}}_{\text{pos}} \in \mathbb{R}^{T\times 1\times C_{\text{f}}}$ to it. We then use the sequence-to-sequence model DSTformer to calculate $\mathbf{F}^i \in \mathbb{R}^{T\times J\times C_{\text{f}}}$ $(i = 1,\dots,N)$ where N is the network depth. We apply a linear layer with tanh activation [29] to $\mathbf{F}^N$ to compute the motion representation $\mathbf{E} \in \mathbb{R}^{T\times J\times C_{\text{e}}}$. Finally, we apply a linear transformation to $\mathbf{E}$ to estimate 3D motion $\hat{\mathbf{X}} \in \mathbb{R}^{T\times J\times C_{\text{out}}}$. Here, T denotes the sequence length, and J denotes the number of body joints. C_{in}, C_{f}, C_{e}, and C_{out} denote the channel numbers of input, feature, embedding, and output respectively. We first introduce the basic building blocks of DSTformer, *i.e.* Spatial and Temporal Blocks with Multi-Head Self-Attention (MHSA), and then explain the DSTformer architecture design.

Spatial Block. Spatial MHSA (S-MHSA) aims at modeling the relationship among the joints within the same time step. It is defined as

$$\begin{aligned}\text{S-MHSA}(\mathbf{Q}_{\text{S}}, \mathbf{K}_{\text{S}}, \mathbf{V}_{\text{S}}) &= [\text{head}_1; ...; \text{head}_h]\mathbf{W}^P_{\text{S}},\\ \text{head}_i &= \text{softmax}(\frac{\mathbf{Q}^i_{\text{S}}(\mathbf{K}^i_{\text{S}})'}{\sqrt{d_K}})\mathbf{V}^i_{\text{S}},\end{aligned} \quad (1)$$

where $\mathbf{W}^P_{\text{S}}$ is a projection parameter matrix, h is the number of the heads, $i \in 1,\dots,h$, and $'$ denotes matrix transpose. We utilize self-attention to get the query $\mathbf{Q}_{\text{S}}$, key $\mathbf{K}_{\text{S}}$, and value $\mathbf{V}_{\text{S}}$ from input per-frame spatial feature $\mathbf{F}_{\text{S}} \in \mathbb{R}^{J\times C_{\text{e}}}$ for each head$_i$,

$$\mathbf{Q}^i_{\text{S}} = \mathbf{F}_{\text{S}}\mathbf{W}^{(Q,i)}_{\text{S}},\ \mathbf{K}^i_{\text{S}} = \mathbf{F}_{\text{S}}\mathbf{W}^{(K,i)}_{\text{S}},\ \mathbf{V}^i_{\text{S}} = \mathbf{F}_{\text{S}}\mathbf{W}^{(V,i)}_{\text{S}}, \quad (2)$$

where $\mathbf{W}^{(Q,i)}_{\text{S}}$, $\mathbf{W}^{(K,i)}_{\text{S}}$, $\mathbf{W}^{(V,i)}_{\text{S}}$ are projection matrices, and d_K is the feature dimension of $\mathbf{K}_{\text{S}}$. We apply S-MHSA to features of different time steps in parallel. Residual connection and layer normalization (LayerNorm) are used to the S-MHSA result, which is further fed into a multilayer perceptron (MLP), and followed by a residual connection and LayerNorm following [105]. We denote the entire spatial block with MHSA, LayerNorm, MLP, and residual connections by $\mathcal{S}$.

Temporal Block. Temporal MHSA (T-MHSA) aims at modeling the relationship across the time steps for a body joint. Its computation process is similar with S-MHSA except that the MHSA is applied to the per-joint temporal feature $\mathbf{F}_\text{T} \in \mathbb{R}^{T \times C_e}$ and parallelized over the spatial dimension.

$$\begin{aligned} \text{T-MHSA}(\mathbf{Q}_\text{T}, \mathbf{K}_\text{T}, \mathbf{V}_\text{T}) &= [\text{head}_1; ...; \text{head}_h]\mathbf{W}_\text{T}^P, \\ \text{head}_i &= \text{softmax}(\frac{\mathbf{Q}_\text{T}^i(\mathbf{K}_\text{T}^i)'}{\sqrt{d_K}})\mathbf{V}_\text{T}^i, \end{aligned} \tag{3}$$

where $i \in 1, \ldots, h$, $\mathbf{Q}_\text{T}$, $\mathbf{K}_\text{T}$, $\mathbf{V}_\text{T}$ are computed similar with Formula 2. We denote the entire temporal block by $\mathcal{T}$.

Dual-stream Spatio-temporal Transformer. Given spatial and temporal MHSA that captures the intra-frame and inter-frame body joint interactions respectively, we assemble the basic building blocks to fuse the spatial and temporal information in the flow. We design a dual-stream architecture with the following assumptions: 1) Both streams should be capable of modeling the comprehensive spatio-temporal context. 2) Each stream should be specialized in different spatio-temporal aspects. 3) The two streams should be fused together, with the fusion weights dynamically balanced depending on the input spatio-temporal characteristics.

Hence, we stack the spatial and temporal MHSA blocks in different orders, forming two parallel computation branches. The output features of the two branches are fused using adaptive weights predicted by an attention regressor. The dual-stream-fusion module is then repeated for N times:

$$\mathbf{F}^i = \boldsymbol{\alpha}_\text{ST}^i \circ \mathcal{T}_1^i(\mathcal{S}_1^i(\mathbf{F}^{i-1})) + \boldsymbol{\alpha}_\text{TS}^i \circ \mathcal{S}_2^i(\mathcal{T}_2^i(\mathbf{F}^{i-1})), \quad i \in 1, \ldots, N, \tag{4}$$

where $\mathbf{F}^i$ denotes the feature embedding at depth i, $\circ$ denotes element-wise production. Orders of $\mathcal{S}$ and $\mathcal{T}$ blocks are shown in Figure 2, and different blocks do not share weights. Adaptive fusion weights $\boldsymbol{\alpha}_\text{ST}, \boldsymbol{\alpha}_\text{TS} \in \mathbb{R}^{N \times T \times J}$ are given by

$$\boldsymbol{\alpha}_\text{ST}^i, \boldsymbol{\alpha}_\text{TS}^i = \text{softmax}(\mathcal{W}([\mathcal{T}_1^i(\mathcal{S}_1^i(\mathbf{F}^{i-1})), \mathcal{S}_2^i(\mathcal{T}_2^i(\mathbf{F}^{i-1}))])), \tag{5}$$

where $\mathcal{W}$ is a learnable linear transformation. $[,]$ denotes concatenation.

3.3. Unified Pretraining

We address two key challenges when designing the unified pretraining framework: 1) How to learn a powerful motion representation with a universal pretext task. 2) How to utilize large-scale but heterogeneous human motion data in all kinds of formats.

For the first challenge, we follow the successful practices in language [11, 29, 84] and vision [6, 34] modeling to construct the supervision signals, *i.e*. mask part of the input and use the encoded representations to reconstruct the whole input. Note that such "cloze" task naturally exists in human motion analysis, that is to recover the lost depth information from the 2D visual observations, *i.e*. 3D human pose estimation. Inspired by this, we leverage the large-scale 3D mocap data [71] and design a 2D-to-3D lifting pretext task. We first extract the 2D skeleton sequences $\mathbf{x}$ by projecting the 3D motion orthographically. Then, we corrupt $\mathbf{x}$ by randomly masking and adding noise to produce the corrupted 2D skeleton sequences, which also resemble the 2D detection results as it contains occlusions, detection failures, and errors. Both joint-level and frame-level masks are applied with certain probabilities. We use the aforementioned motion encoder to get motion representation $\mathbf{E}$ and reconstruct 3D motion $\hat{\mathbf{X}}$. We then compute the joint loss $\mathcal{L}_\text{3D}$ between $\hat{\mathbf{X}}$ and GT 3D motion $\mathbf{X}$. We also add the velocity loss $\mathcal{L}_\text{O}$ following previous works [83, 126]. The 3D reconstruction losses are thus given by

$$\mathcal{L}_\text{3D} = \sum_{t=1}^{T}\sum_{j=1}^{J} \parallel \hat{\mathbf{X}}_{t,j} - \mathbf{X}_{t,j} \parallel_2, \quad \mathcal{L}_\text{O} = \sum_{t=2}^{T}\sum_{j=1}^{J} \parallel \hat{\mathbf{O}}_{t,j} - \mathbf{O}_{t,j} \parallel_2, \tag{6}$$

where $\hat{\mathbf{O}}_t = \hat{\mathbf{X}}_t - \hat{\mathbf{X}}_{t-1}$, $\mathbf{O}_t = \mathbf{X}_t - \mathbf{X}_{t-1}$.

For the second challenge, we notice that 2D skeletons could serve as a universal medium as they can be extracted from all sorts of motion data sources. We further incorporate in-the-wild RGB videos into the 2D-to-3D lifting framework for unified pretraining. For RGB videos, the 2D skeletons $\mathbf{x}$ could be given by manual annotation [3] or 2D pose estimators [13, 96], and the depth channel of the extracted 2D skeletons is intrinsically "masked". Similarly, we add extra masks and noises to degrade $\mathbf{x}$ (if $\mathbf{x}$ already contains detection noise, only masking is applied). As 3D motion GT $\mathbf{X}$ is not available for these data, we apply a weighted 2D re-projection loss which is calculated by

$$\mathcal{L}_\text{2D} = \sum_{t=1}^{T}\sum_{j=1}^{J} \boldsymbol{\delta}_{t,j} \|\hat{\mathbf{x}}_{t,j} - \mathbf{x}_{t,j}\|_2, \tag{7}$$

where $\hat{\mathbf{x}}$ is the 2D orthographical projection of the estimated 3D motion $\hat{\mathbf{X}}$, and $\boldsymbol{\delta} \in \mathbb{R}^{T \times J}$ is given by visibility annotation or 2D detection confidence.

The total pretraining loss is computed by

$$\mathcal{L} = \underbrace{\mathcal{L}_\text{3D} + \lambda_\text{O}\mathcal{L}_\text{O}}_{\text{for 3D data}} + \underbrace{\mathcal{L}_\text{2D}}_{\text{for 2D data}}, \tag{8}$$

where λ_O is a constant coefficient to balance the losses.

3.4. Task-specific Finetuning

The learned feature embedding $\mathbf{E}$ serves as a 3D-aware and temporal-aware human motion representation. For downstream tasks, we adopt the *minimalist* design principle, *i.e*. implementing a shallow downstream network and training without bells and whistles. In practice, we use an extra linear layer or an MLP with one hidden layer. We then finetune the whole network end-to-end.

3D Pose Estimation. As we utilize 2D-to-3D lifting as the pretext task, we simply reuse the whole pretrained network. During finetuning, the input 2D skeletons are estimated from videos without extra masks or noises.

Skeleton-based Action Recognition. We directly apply a global average pooling over different persons and timesteps. The result is then fed into an MLP with one hidden layer. The network is trained with cross-entropy classification loss. For one-shot learning, we apply a linear layer after the pooled features to extract clip-level action representation. We introduce the detailed setup of one-shot learning in Section 4.4.

Human Mesh Recovery. We use SMPL [66] model to represent the human mesh and regress its parameters. The SMPL model consists of pose parameters $\theta \in \mathbb{R}^{72}$ and shape parameters $\beta \in \mathbb{R}^{10}$, and calculates the 3D mesh as $\mathcal{M}(\theta, \beta) \in \mathbb{R}^{6890\times 3}$. To regress the pose parameters for each frame, we feed the motion embeddings $\mathbf{E}$ to an MLP with one hidden layer and get $\hat{\boldsymbol{\theta}} \in \mathbb{R}^{T\times 72}$. To estimate shape parameters, considering that the human shape over a video sequence is supposed to be consistent, we first perform an average pooling of $\mathbf{E}$ over the temporal dimension and then feed it into another MLP to regress a single $\hat{\beta}$ and then expand it to the entire sequence as $\hat{\boldsymbol{\beta}} \in \mathbb{R}^{T\times 10}$. The shape MLP has the same architecture as the pose regression one, and they are initialized with the mean shape and pose, respectively, as in [42]. The overall loss is computed as

$$\mathcal{L} = \lambda_{\text{3D}}^{\text{m}}\mathcal{L}_{\text{3D}}^{\text{m}} + \lambda_\theta\mathcal{L}_\theta + \lambda_\beta\mathcal{L}_\beta + \lambda_{\text{n}}\mathcal{L}_{\text{norm}} + \lambda_{\text{O}}^{\text{m}}\mathcal{L}_{\text{O}}^{\text{m}}, \quad (9)$$

where each term is calculated as

$$\begin{aligned} &\mathcal{L}_{\text{3D}}^{\text{m}} = \|\hat{\mathbf{X}}^{\text{m}} - \mathbf{X}^{\text{m}}\|_1, \quad \mathcal{L}_\theta = \|\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}\|_1, \quad \mathcal{L}_\beta = \|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}\|_1, \\ &\mathcal{L}_{\text{norm}} = \|\hat{\boldsymbol{\theta}}\|_2 + \|\hat{\boldsymbol{\beta}}\|_2, \quad \mathcal{L}_{\text{O}}^{\text{m}} = \|\hat{\mathbf{O}}^{\text{m}} - \mathbf{O}^{\text{m}}\|_2. \end{aligned} \quad (10)$$

Note that each 3D pose in motion $\mathbf{X}^{\text{m}}$ at frame t is regressed from mesh vertices by $\mathbf{X}_t^{\text{m}} = \mathbf{J}\mathcal{M}(\boldsymbol{\theta}_t, \boldsymbol{\beta}_t)$, where $\mathbf{J} \in \mathbb{R}^{J\times 6890}$ is a pre-defined matrix [9]. $\mathbf{O}^{\text{m}} = \mathbf{X}_{t+1}^{\text{m}} - \mathbf{X}_t^{\text{m}}$, $\hat{\mathbf{O}}^{\text{m}} = \hat{\mathbf{X}}_{t+1}^{\text{m}} - \hat{\mathbf{X}}_t^{\text{m}}$. $\lambda_{\text{3D}}^{\text{m}}$, λ_θ, λ_β, λ_{n} and $\lambda_{\text{O}}^{\text{m}}$ are constant coefficients to balance the training loss.

4. Experiments

4.1. Implementation

We implement the proposed motion encoder DSTformer with depth $N = 5$, number of heads $h = 8$, feature size $C_{\text{f}} = 512$, embedding size $C_{\text{e}} = 512$. For pretraining, we use sequence length $T = 243$. The pretrained model could handle different input lengths thanks to the Transformer-based backbone. During finetuning, we set the backbone learning rate to be $0.1\times$ of the new layer learning rate. We introduce the experiment datasets in the following sections respectively. Please refer to the appendix for more experimental details.

4.2. Pretraining

We collect diverse and realistic 3D human motion from two datasets, Human3.6M [36] and AMASS [71]. Human3.6M [36] is a commonly used indoor dataset for 3D human pose estimation which contains 3.6 million video frames of professional actors performing daily actions. Following previous works [73, 83], we use subjects 1, 5, 6, 7, 8 for training, and subjects 9, 11 for testing. AMASS [71] integrates most existing marker-based Mocap datasets [1,2,4,10,14,17,32,35,48,65,67,72,79,90,103,104] and parameterizes them with a common representation. We do not use the images or 2D detection results of the two datasets during pretraining as Mocap datasets usually do not provide raw videos. Instead, we use orthographic projection to get the uncorrupted 2D skeletons. We further incorporate two in-the-wild RGB video datasets PoseTrack [3] (annotated) and InstaVariety [39] (unannotated) for higher motion diversity. We align the body keypoint definitions with Human3.6M and calibrate the camera coordinates to pixel coordinates following [26]. We randomly zero out 15% joints, and sample noises from a mixture of Gaussian and uniform distributions [16]. We first train on 3D data only for 30 epochs, then train on both 3D data and 2D data for 60 epochs, following the curriculum learning practices [8, 111].

4.3. 3D Pose Estimation

We evaluate the 3D pose estimation performance on Human3.6M [36] and report the mean per joint position error (MPJPE) in millimeters, which measures the average distance between the predicted joint positions and the GT after aligning the root joint. We also compute the mean per-joint velocity error (MPJVE) to evalute the temporal smoothness following previous works [126, 127]. We use the Stacked Hourglass (SH) networks [81] to extract the 2D skeletons from videos, and finetune the entire network on Human3.6M [36] training set. In addition, we train a separate model of the same architecture, but with random initialization rather than pretrained weights. As shown in Table 1 (top), the model trained from scratch outperforms

Method	T	Dire.	Disc.	Eat	Greet	Phone	Photo	Pose	Purch.	Sit	SitD	Smoke	Wait	WalkD	Walk	WalkT	Avg
Martinez *et al.* [73] ICCV'17	1	51.8	56.2	58.1	59.0	69.5	78.4	55.2	58.1	74.0	94.6	62.3	59.1	65.1	49.5	52.4	62.9
Pavlakos *et al.* [82] CVPR'18	1	48.5	54.4	54.4	52.0	59.4	65.3	49.9	52.9	65.8	71.1	56.6	52.9	60.9	44.7	47.8	56.2
LCN [27] ICCV'19	1	46.8	52.3	44.7	50.4	52.9	68.9	49.6	46.4	60.2	78.9	51.2	50.0	54.8	40.4	43.3	52.7
Xu *et al.* [114] CVPR'21	1	45.2	49.9	47.5	50.9	54.9	66.1	48.5	46.3	59.7	71.5	51.4	48.6	53.9	39.9	44.1	51.9
VideoPose3D [83] CVPR'19	243	45.2	46.7	43.3	45.6	48.1	55.1	44.6	44.3	57.3	65.8	47.1	44.0	49.0	32.8	33.9	46.8
Cai *et al.* [12] ICCV'19	7	44.6	47.4	45.6	48.8	50.8	59.0	47.2	43.9	57.9	61.9	49.7	46.6	51.3	37.1	39.4	48.8
Yeh *et al.* [122] NeurIPS'19	243	44.8	46.1	43.3	46.4	49.0	55.2	44.6	44.0	58.3	62.7	47.1	43.9	48.6	32.7	33.3	46.7
Liu *et al.* [63] CVPR'20	243	41.8	44.8	41.1	44.9	47.4	54.1	43.4	42.2	56.2	63.6	45.3	43.5	45.3	31.3	32.2	45.1
* Cheng *et al.* [21] AAAI'20	128	36.2	38.1	42.7	35.9	**38.2**	45.7	36.8	42.0	**45.9**	**51.3**	41.8	41.5	43.8	33.1	28.6	40.1
* UGCN [109] ECCV'20	96	38.2	41.0	45.9	39.7	41.4	51.4	41.6	41.4	52.0	57.4	41.8	44.4	41.6	33.1	30.0	42.6
† PoseFormer [127] ICCV'21	81	41.5	44.8	39.8	42.5	46.5	51.6	42.1	42.0	53.3	60.7	45.5	43.3	46.1	31.8	32.2	44.3
* Wehrbein *et al.* [112] ICCV'21	200	38.5	42.5	39.9	41.7	46.5	51.6	39.9	40.8	49.5	56.8	45.3	46.4	46.8	37.8	40.4	44.3
† MHFormer [52] CVPR'22	351	39.2	43.1	40.1	40.9	44.9	51.2	40.6	41.3	53.5	60.3	43.7	41.1	43.8	29.8	30.6	43.0
*† MixSTE [126] CVPR'22	243	36.7	39.0	36.5	39.4	40.2	**44.9**	39.8	36.9	47.9	54.8	39.6	37.8	39.3	29.7	30.6	39.8
† P-STMO [87] ECCV'22	243	38.4	42.1	39.8	40.2	45.2	48.9	40.4	38.3	53.8	57.3	43.9	41.6	42.2	29.3	29.3	42.1
† Ours (scratch)	243	36.3	38.7	38.6	33.6	42.1	50.1	36.2	35.7	50.1	56.6	41.3	37.4	37.7	25.6	26.5	39.2
† Ours (finetune)	243	**36.1**	**37.5**	**35.8**	**32.1**	40.3	46.3	**36.1**	**35.3**	46.9	53.9	**39.5**	**36.3**	**35.8**	**25.1**	**25.3**	**37.5**
Method	T	Dire.	Disc.	Eat	Greet	Phone	Photo	Pose	Purch.	Sit	SitD	Smoke	Wait	WalkD	Walk	WalkT	Avg
Martinez *et al.* [73] ICCV'17	1	37.7	44.4	40.3	42.1	48.2	54.9	44.4	42.1	54.6	58.0	45.1	46.4	47.6	36.4	40.4	45.5
LCN [27] ICCV'19	1	36.3	38.8	29.7	37.8	34.6	42.5	39.8	32.5	36.2	39.5	34.4	38.4	38.2	31.3	34.2	36.3
Xu *et al.* [114] CVPR'21	1	35.8	38.1	31.0	35.3	35.8	43.2	37.3	31.7	38.4	45.5	35.4	36.7	36.8	27.9	30.7	35.8
UGCN [109] ECCV'20	96	23.0	25.7	22.8	22.6	24.1	30.6	24.9	24.5	31.1	35.0	25.6	24.3	25.1	19.8	18.4	25.6
† PoseFormer [127] ICCV'21	81	30.0	33.6	29.9	31.0	30.2	33.3	34.8	31.4	37.8	38.6	31.7	31.5	29.0	23.3	23.1	31.3
† MHFormer [52] CVPR'22	351	27.7	32.1	29.1	28.9	30.0	33.9	33.0	31.2	37.0	39.3	30.0	31.0	29.4	22.2	23.0	30.5
† MixSTE [126] CVPR'22	243	21.6	22.0	20.4	21.0	20.8	24.3	24.7	21.9	26.9	24.9	21.2	21.5	20.8	14.7	15.6	21.6
† P-STMO [87] ECCV'22	243	28.5	30.1	28.6	27.9	29.8	33.2	31.3	27.8	36.0	37.4	29.7	29.5	28.1	21.0	21.0	29.3
† Ours (scratch)	243	16.7	19.9	17.1	16.5	17.4	18.8	19.3	20.5	24.0	22.1	18.6	**16.8**	16.7	10.8	11.5	17.8
† Ours (finetune)	243	**15.9**	**17.3**	**16.9**	**14.6**	**16.8**	**18.6**	**18.6**	**18.4**	**22.0**	**21.8**	**17.3**	16.9	**16.1**	**10.5**	**11.4**	**16.9**
Method	T	Dire.	Disc.	Eat	Greet	Phone	Photo	Pose	Purch.	Sit	SitD	Smoke	Wait	WalkD	Walk	WalkT	Avg
VideoPose3D [83] CVPR'19	243	3.0	3.1	2.2	3.4	2.3	2.7	2.7	3.1	2.1	2.9	2.3	2.4	3.7	3.1	2.8	2.8
† PoseFormer [127] ICCV'21	81	3.2	3.4	2.6	3.6	2.6	3.0	2.9	3.2	2.6	3.3	2.7	2.7	3.8	3.2	2.9	3.1
*† MixSTE [126] CVPR'22	243	2.5	2.7	1.9	2.8	1.9	2.2	2.3	2.6	1.6	2.2	1.9	2.0	3.1	2.6	2.2	2.3
† Ours (scratch)	243	1.8	2.1	1.5	2.0	1.5	1.9	1.8	2.1	1.2	1.8	1.5	1.4	2.6	2.0	1.7	1.8
† Ours (finetune)	243	**1.7**	**1.9**	**1.4**	**1.9**	**1.4**	**1.7**	**1.7**	**1.9**	**1.1**	**1.6**	**1.4**	**1.3**	**2.4**	**1.9**	**1.6**	**1.7**

Table 1. **Quantitative comparison of 3D human pose estimation on Human3.6M.** (Top) MPJPE (mm) using detected 2D pose sequences. (Middle) MPJPE (mm) using GT 2D pose sequences. (Bottom) MPJVE (mm) using detected 2D pose sequences. T denotes the clip length used by the method. We select the best results reported by each work. * denotes using HRNet [96] for 2D detection. † denotes implemented with a spatio-temporal Transformer design. The best and second-best results are highlighted in bold and underlined formats.

previous methods including other Transformer-based designs with spatio-temporal modeling. It shows the effectiveness of the proposed DSTformer in terms of learning 3D geometric structures and temporal dynamics. To further evaluate the upper bound of the models' capability, we compare the performance when using 2D GT pose sequences as input, which gets rid of the influence of different 2D detectors. As shown in Table 1 (middle), our models significantly outperform all the previous approaches. Table 1 (bottom) shows that both of our models also surpass previous works in terms of MPJVE, implying better temporal coherence. We attribute the performance advantage of our scratch model to the proposed DSTformer design. We include more comparisons and analysis to demonstrate the advantage of DSTformer with regard to other spatio-temporal architectures in Section 4.6 and supplementary materials. Additionally, our method achieves lower errors with the proposed pretraining stage.

4.4. Skeleton-based Action Recognition

We further explore the possibility to learn action semantics with the pretrained human motion representations. We use the human action dataset NTU-RGB+D [86] which contains 57K videos of 60 action classes, and we follow the data splits Cross-subject (X-Sub) and Cross-view (X-View). The dataset has an extended version, NTU-RGB+D-120 [60], which contains 114K videos of 120 action classes. We follow the suggested *One-shot* action recognition protocol on NTU-RGB+D-120. For both datasets, we use HRNet [96] to extract 2D skeletons following [30]. Similarly, we train a scratch model with random initialization for comparison. As Table 2 (left) shows, our methods are comparable or superior to the state-of-the-art approaches. Notably, the pretraining stage accounts for a large performance gain.

Additionally, we delve into the one-shot setting which holds significant practical importance. Real-world applications often require fine-grained action recognition in specific domains such as education, sports, and healthcare. Unfortunately, the action classes in these scenarios are not typically defined in public datasets. As a result, only limited annotations for these novel action classes are available, making accurate recognition a challenging task. As proposed in [60], we report the results on the evaluation set of 20 novel classes

Method	X-Sub	X-View
ST-GCN [116] AAAI'18	81.5	88.3
2s-AGCN [89] CVPR'19	88.5	95.1
MS-G3D [64] CVPR'20	91.5	96.2
Shift-GCN [20] CVPR'20	90.7	96.5
CrosSCLR [58] CVPR'21	86.2	92.5
MCC (finetune) [94] ICCV'21	89.7	96.3
SCC (finetune) [118] ICCV'21	88.0	94.9
UNIK (finetune) [117] BMVC'21	86.8	94.4
CTR-GCN [19] ICCV'21	92.4	96.8
PoseConv3D [30] CVPR'22	**93.1**	95.7
Ours (scratch)	87.7	94.1
Ours (finetune)	93.0	**97.2**

Method	Accuracy
ST-LSTM + AvgPool [61]	42.9
ST-LSTM + FC [62]	42.1
ST-LSTM + Attention [62]	41.0
APSR [60]	45.3
TCN OneShot [85]	46.5
SL-DML [76]	50.9
Skeleton-DML [75]	54.2
Ours (scratch)	61.0
Ours (finetune)	**67.4**

Table 2. **Quantitative comparison of skeleton-based action recognition accuracy.** (Left) Cross-subject and cross-view recognition accuracy on NTU-RGB+D. All the methods are evaluated using only the "joint" modality with 1-clip sampling for the fairness of comparison. (Right) One-shot recognition accuracy on NTU-RGB+D-120. All results are top-1 accuracy (%).

Method	Input	T	Human3.6M			3DPW		
			MPVE↓	MPJPE↓	PA-MPJPE↓	MPVE↓	MPJPE↓	PA-MPJPE↓
HMR [38] CVPR'18	image	1	-	88.0	56.8	-	130.0	81.3
† SPIN [44] ICCV'19	image	1	82.3	59.4	39.3	129.1	100.9	59.1
Pose2Mesh [24] ECCV'20	2D pose	1	85.3	64.9	48.7	109.3	91.4	60.1
I2L-MeshNet [78] ECCV'20	image	1	-	55.7	41.7	110.1	93.2	58.6
† HybrIK [50] CVPR'21	image	1	58.1	47.4	30.1	82.4	71.3	41.9
METRO [54] CVPR'21	image	1	-	54.0	36.7	88.2	77.1	47.9
Mesh Graphormer [55] ICCV'21	image	1	-	51.2	34.5	87.7	74.7	45.6
PARE [43] ICCV'21	image	1	-	-	-	88.6	74.5	46.5
ROMP [98] ICCV'21	image	1	-	-	-	108.3	91.3	54.9
PyMAF [125] ICCV'21	image	1	-	57.7	40.5	110.1	92.8	58.9
ProHMR [46] ICCV'21	image	1	-	-	41.2	-	-	59.8
OCHMR [40] CVPR'22	image	1	-	-	-	107.1	89.7	58.3
3DCrowdNet [25] CVPR'22	image	1	-	-	-	98.3	81.7	51.5
CLIFF [53] ECCV'22	image	1	-	47.1	32.7	81.2	69.0	43.0
FastMETRO [22] ECCV'22	image	1	-	52.2	33.7	84.1	73.5	44.6
VisDB [121] ECCV'22	image	1	-	51.0	34.5	85.5	73.5	44.9
TemporalContext [5] CVPR'19	video	32	-	77.8	54.3	-	-	72.2
HMMR [39] CVPR'19	video	20	-	-	56.9	139.3	116.5	72.6
DSD-SATN [99] ICCV'19	video	9	-	59.1	42.4	-	-	69.5
VIBE [42] CVPR'20	video	16	-	65.6	41.4	99.1	82.9	51.9
TCMR [23] CVPR'21	video	16	-	62.3	41.1	102.9	86.5	52.7
† MAED [107] ICCV'21	video	16	84.1	60.4	38.3	93.3	79.0	45.7
MPS-Net [113] CVPR'22	video	16	-	69.4	47.4	99.7	84.3	52.1
* PoseBERT [7] TPAMI'22 (+SPIN [44])	video	16	-	-	-	-	-	57.3 ↓ 2.3
* SmoothNet [123] ECCV'22 (+SPIN [44])	video	32	-	67.5 ↓ 1.0	46.3 ↓ 0.2	-	86.7 ↓ 0.9	52.7 ↓ 0.6
Ours (scratch)	2D motion	16	75.7	62.8	41.0	99.1	85.5	50.2
Ours (finetune)	2D motion	16	65.5	53.8	34.9	88.1	76.9	47.2
Ours (finetune) + SPIN [44]	video	16	63.7 ↓ 18.6	52.2 ↓ 7.2	35.7 ↓ 3.6	92.8 ↓36.3	79.6 ↓ 21.3	48.2 ↓ 10.9
Ours (finetune) + MAED [107]	video	16	66.8 ↓ 17.3	54.8 ↓ 5.6	36.4 ↓ 1.9	84.4 ↓ 8.9	72.3 ↓ 6.7	42.3 ↓ 3.4
Ours (finetune) + HybrIK [50]	video	16	**52.6** ↓ 5.5	**43.1** ↓ 4.3	**27.8** ↓ 2.3	**79.4** ↓ 3.0	**68.8** ↓ 2.5	**40.6** ↓ 1.3

Table 3. **Quantitative comparison of human mesh recovery on Human3.6M and 3DPW datasets.** T denotes the clip length used by the method. † denotes the results obtained with official model weights. The rest are all officially reported results. The gains in * correspond to different re-implemented SPIN [44] results.

using only 1 labeled video for each class. The auxiliary set contains the other 100 classes, and all samples of these classes can be used. We train the model on the auxiliary set using the supervised contrastive learning technique [41]. For a batch of auxiliary data, samples of the same class are pulled together, while samples of different classes are pushed away in the action embedding space. During the evaluation, we calculate the cosine distance between the test examples and the exemplars, and use 1-nearest neighbor to determine the class. Table 2 (right) illustrates that the proposed models outperform state-of-the-art by a considerable margin. Moreover, it is noteworthy that our pretrained model achieves optimal performance with only 1-2 epochs of fine-tuning. Our results indicate that the pretraining stage is effective in learning a robust motion representation that generalizes well to novel downstream tasks, even with limited data annotations.

4.5. Human Mesh Recovery

We conduct experiments on Human3.6M [36] and 3DPW [106] datasets and additionally add COCO [57] dataset during training following [42, 54, 107]. We keep the same training and test split for both datasets as in [73] (Section 4.2)

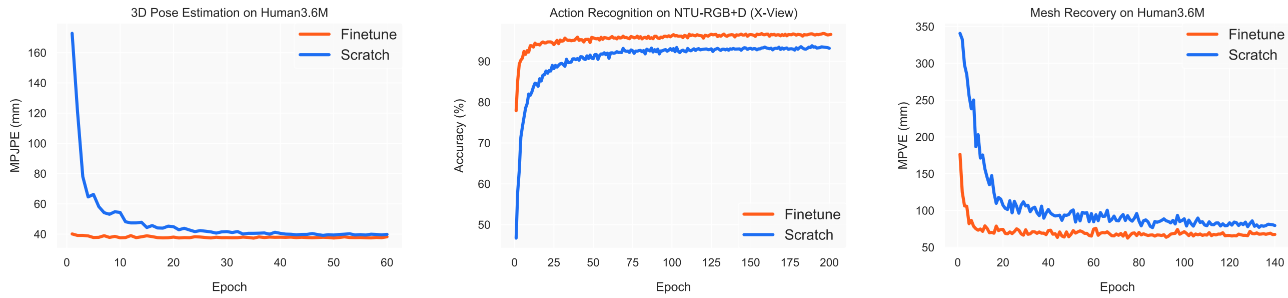

Figure 3. **Learning curves of finetuning and training from scratch.**

Backbone (frozen)	MPJPE ↓ (3D pose)	MPVE ↓ (mesh)	Accuracy ↑ (action x-view)	Accuracy↑ (action 1-shot)
Random	404.4mm	114.4mm	47.6%	46.8%
Pretrained	**40.3mm**	**72.1mm**	**87.3%**	**60.7%**

Table 4. **Comparison of partial finetuning.**

Pretrain	Noise	Mask	2D	MPJPE↓ (3D pose)	MPVE↓ (mesh)	Accuracy↑ (action x-sub)
-	-	-	-	39.2mm	75.7mm	87.7%
✓	-	-	-	38.8mm	70.6mm	89.4%
✓	✓	-	-	38.1mm	68.4mm	90.7%
✓	✓	✓	-	**37.4mm**	67.8mm	91.9%
✓	✓	✓	✓	37.5mm	**65.5mm**	**93.0%**

Table 5. **Comparison of pretraining strategies.**

and [42, 54, 107], respectively. Following the common practice [38,42,45,107], we report MPJPE (mm) and PA-MPJPE (mm) of 14 joints obtained by $\mathbf{J}\mathcal{M}(\theta, \beta)$. PA-MPJPE calculates MPJPE after aligning with GT in translation, rotation, and scale. We further report the mean per vertex error (MPVE) (mm) of the mesh $\mathcal{M}(\theta, \beta)$, which measures the average distance between the estimated and GT vertices after aligning the root joint. Note that most previous works [23, 38, 42, 44, 50, 54, 68, 107] use more datasets other than COCO [57] during training, such as LSP [37], MPI-INF-3DHP [74], *etc*., while we do not. Table 3 demonstrates that our finetuned model delivers competitive results on both Human3.6M and 3DPW datasets, surpassing all the state-of-the-art *video-based* methods, including MAED [107], especially on the MPVE error. Nonetheless, we note that estimating full-body mesh from sparse 2D keypoints alone [9, 24] is an ill-posed problem because it lacks human shape information. In light of this, we propose a hybrid approach that leverages the strengths of both our framework (coherent motion) and RGB-based methods (accurate shape). We introduce a refiner module that can be easily integrated with existing image/video-based methods, similar to [7, 123]. Specifically, our refiner module is an MLP that takes the combination of our pretrained motion representations and an initial prediction, regressing a residual in joint rotations. Our approach effectively improves the state-of-the-art methods [44,50,107] and achieves the lowest error to date.

4.6. Ablation Studies

Finetune vs. Scratch. We compare the training progress of finetuning the pretrained model and training from scratch. As Figure 3 shows, models initialized with pretrained weights demonstrate superior performance and faster convergence on all three tasks. This observation suggests that the pretrained model learns transferable knowledge about human motion, facilitating the learning of multiple downstream tasks.

Partial Finetuning. In addition to end-to-end finetuning, we freeze the motion encoder backbone and only train the regression head for each downstream task. To verify the effectiveness of the pretrained motion representations, we compared the pretrained motion encoder with a randomly initialized motion encoder. We report results of 3D pose and mesh on Human3.6M, action on NTU-RGB+D and NTU-RGB+D-120 (same for the tables below). It can be seen in Table 4 that based on the frozen pretrained motion representations, our method still achieves competitive performance on multiple downstream tasks and shows a large improvement compared to the baseline. Pretraining and partial finetuning make it possible for all the downstream tasks to share the same backbone, significantly reducing computation overhead for applications requiring multi-task inference.

Pretraining Strategies. We evaluate how different pretraining strategies influence the performance of downstream tasks. Starting from the scratch baseline, we apply the proposed strategies one by one. As shown in Table 5, a vanilla 2D-to-3D pretraining stage brings benefits to all the downstream tasks. Introducing corruptions additionally improves the learned motion embeddings. Unified pretraining with in-the-wild videos (*w.* 2D) enjoys higher motion diversity, which further helps several downstream tasks.

Pretraining with Different Backbones. We further study the universality of the proposed pretraining approach. We

Setting	MPJPE ↓ (3D pose)	MPVE ↓ (mesh)	Accuracy ↑ (action x-view)	Accuracy↑ (action 1-shot)
TCN (scratch)	50.1mm	92.6mm	91.5%	52.4%
TCN (finetune)	**47.9mm**	**86.3mm**	**92.8%**	**59.9%**
PoseFormer (scratch)	44.8mm	85.9mm	94.2%	57.4%
PoseFormer (finetune)	**41.5mm**	**80.5mm**	**95.9%**	**60.7%**

Table 6. **Comparison of different backbones.**

Arch.	(a)	(b)	(c)	(d)	(e)	(f)
Design	S-T	T-S	S + T	ST-MHSA	S-T + T-S (Average)	S-T + T-S (Adaptive)
MPJPE ↓	$40.58_{\pm 0.31}$	$41.05_{\pm 0.24}$	$41.76_{\pm 0.22}$	$41.54_{\pm 0.35}$	$39.87_{\pm 0.32}$	$\mathbf{39.25}_{\pm 0.27}$

Table 7. **Comparison of model architecture variants.** All the methods are trained on Human3.6M from scratch over 5 runs and measured by MPJPE (mm) with mean and standard deviation.

replace the motion encoder backbone with two variants: TCN [83] and PoseFormer [127]. The models are slightly modified to a *seq2seq* version, while all the configurations for pretraining and finetuning are simply followed. Table 6 shows that the proposed approach consistently benefits different backbone models on different tasks.

Model Architecture. Finally, we study the design choices of DSTformer. From (a) to (f) in Table 7, we compare different structure designs of the basic Transformer module. (a) and (b) are single-stream versions with different orders. (a) is conceptually similar to PoseFormer [127], MHFormer [52], and MixSTE [126]. (c) limits each stream to either temporal or spatial modeling before fusion and is similar to MAED [107]. (d) directly connects S-MHSA and T-MHSA without the MLP in between and is similar to the *MSA-T* variant in MAED [107]. (e) replaces the adaptive fusion with average pooling on two streams. (f) is the proposed DSTformer design. The result statistically confirms our design principles that both streams should be capable and meanwhile complementary, as introduced in Section 3.2. In addition, we find out that pairing each self-attention block with an MLP is crucial, as it could project the learned feature interactions and bring nonlinearity. In general, we design the model architecture for the 3D pose estimation task and apply it to all other tasks without additional adjustment.

5. Conclusion

In this work, we provide a unified perspective to tackling various human-centric video tasks. We develop a pretraining approach to learn human motion representations from large-scale and heterogeneous data sources. We also propose DSTformer as a universal human motion encoder. Experimental results on multiple benchmarks demonstrate the versatility of the learned motion representations. Future work could explore fusing the learned motion representations with generic video architectures as a human-centric semantic feature and applying it to more tasks (*e.g.*, action assessment, segmentation).

Acknowledgement This work was partially supported by MOST-2022ZD0114900. We would like to thank Hai Ci and Jiefeng Li for their exceptional support.

References

[1] Advanced Computing Center for the Arts and Design. ACCAD MoCap Dataset.

[2] Ijaz Akhter and Michael J. Black. Pose-conditioned joint angle limits for 3D human pose reconstruction. In *CVPR*, pages 1446–1455, June 2015.

[3] Mykhaylo Andriluka, Umar Iqbal, Eldar Insafutdinov, Leonid Pishchulin, Anton Milan, Juergen Gall, and Bernt Schiele. Posetrack: A benchmark for human pose estimation and tracking. In *CVPR*, pages 5167–5176, 2018.

[4] Andreas Aristidou, Ariel Shamir, and Yiorgos Chrysanthou. Digital dance ethnography: Organizing large dance collections. *JOCCH*, 12(4):1–27, 2019.

[5] Anurag Arnab, Carl Doersch, and Andrew Zisserman. Exploiting temporal context for 3d human pose estimation in the wild. In *CVPR*, pages 3395–3404, 2019.

[6] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. BEit: BERT pre-training of image transformers. In *ICLR*, 2022.

[7] Fabien Baradel, Romain Brégier, Thibault Groueix, Philippe Weinzaepfel, Yannis Kalantidis, and Grégory Rogez. Posebert: A generic transformer module for temporal 3d human modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[8] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *ICML*, page 41–48, 2009.

[9] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J Black. Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In *ECCV*, pages 561–578, 2016.

[10] Federica Bogo, Javier Romero, Gerard Pons-Moll, and Michael J Black. Dynamic faust: Registering human bodies in motion. In *CVPR*, pages 6233–6242, 2017.

[11] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *NeurIPS*, 33:1877–1901, 2020.

[12] 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.

[13] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, pages 7291–7299, 2017.

[14] Carnegie Mellon University. CMU MoCap Dataset.

[15] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, pages 6299–6308, 2017.

[16] Ju Yong Chang, Gyeongsik Moon, and Kyoung Mu Lee. Poselifter: Absolute 3d human pose lifting network from a single noisy 2d human pose. *arXiv preprint arXiv:1910.12029*, 2019.

[17] Anargyros Chatzitofis, Leonidas Saroglou, Prodromos Boutis, Petros Drakoulis, Nikolaos Zioulis, Shishir Subramanyam, Bart Kevelham, Caecilia Charbonnier, Pablo Cesar, Dimitrios Zarpalas, et al. Human4d: A human-centric multimodal dataset for motions and immersive media. *IEEE Access*, 8:176241–176262, 2020.

[18] Xipeng Chen, Kwan-Yee Lin, Wentao Liu, Chen Qian, and Liang Lin. Weakly-supervised discovery of geometry-aware representation for 3d human pose estimation. In *CVPR*, 2019.

[19] Yuxin Chen, Ziqi Zhang, Chunfeng Yuan, Bing Li, Ying Deng, and Weiming Hu. Channel-wise topology refinement graph convolution for skeleton-based action recognition. In *ICCV*, 2021.

[20] Ke Cheng, Yifan Zhang, Xiangyu He, Weihan Chen, Jian Cheng, and Hanqing Lu. Skeleton-based action recognition with shift graph convolutional network. In *CVPR*, 2020.

[21] Yu Cheng, Bo Yang, Bo Wang, and Robby T Tan. 3d human pose estimation using spatio-temporal networks with explicit occlusion training. In *AAAI*, 2020.

[22] Junhyeong Cho, Kim Youwang, and Tae-Hyun Oh. Cross-attention of disentangled modalities for 3d human mesh recovery with transformers. In *European Conference on Computer Vision (ECCV)*, 2022.

[23] Hongsuk Choi, Gyeongsik Moon, Ju Yong Chang, and Kyoung Mu Lee. Beyond static features for temporally consistent 3d human pose and shape from a video. In *CVPR*, pages 1964–1973, 2021.

[24] Hongsuk Choi, Gyeongsik Moon, and Kyoung Mu Lee. Pose2mesh: Graph convolutional network for 3d human pose and mesh recovery from a 2d human pose. In *ECCV*, pages 769–787, 2020.

[25] Hongsuk Choi, Gyeongsik Moon, JoonKyu Park, and Kyoung Mu Lee. Learning to estimate robust 3d human mesh from in-the-wild crowded scenes. In *CVPR*, pages 1475–1484, June 2022.

[26] Hai Ci, Xiaoxuan Ma, Chunyu Wang, and Yizhou Wang. Locally connected network for monocular 3d

human pose estimation. *IEEE TPAMI*, pages 1–1, 2020.

[27] Hai Ci, Chunyu Wang, Xiaoxuan Ma, and Yizhou Wang. Optimizing network structure for 3d human pose estimation. In *ICCV*, pages 2262–2271, 2019.

[28] Hai Ci, Mingdong Wu, Wentao Zhu, Xiaoxuan Ma, Hao Dong, Fangwei Zhong, and Yizhou Wang. Gf-pose: Learning 3d human pose prior with gradient fields. *CVPR*, 2023.

[29] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019.

[30] Haodong Duan, Yue Zhao, Kai Chen, Dian Shao, Dahua Lin, and Bo Dai. Revisiting skeleton-based action recognition. In *CVPR*, 2022.

[31] Mark Endo, Kathleen L. Poston, Edith V. Sullivan, Li Fei-Fei, Kilian M. Pohl, and Ehsan Adeli. Gaitforemer: Self-supervised pre-training of transformers via human motion forecasting for few-shot gait impairment severity estimation. In *MICCAI*, pages 130–139, 2022.

[32] Saeed Ghorbani, Kimia Mahdaviani, Anne Thaler, Konrad Kording, Douglas James Cook, Gunnar Blohm, and Nikolaus F. Troje. MoVi: A large multipurpose motion and video dataset, 2020.

[33] Rıza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild. In *CVPR*, pages 7297–7306, 2018.

[34] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross B. Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022.

[35] Ludovic Hoyet, Kenneth Ryall, Rachel McDonnell, and Carol O'Sullivan. Sleight of hand: Perception of finger motion from reduced marker sets. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, 2012.

[36] 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*, 2014.

[37] Sam Johnson and Mark Everingham. Clustered pose and nonlinear appearance models for human pose estimation. In *BMVC*, volume 2, page 5, 2010.

[38] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *CVPR*, pages 7122–7131, 2018.

[39] Angjoo Kanazawa, Jason Y Zhang, Panna Felsen, and Jitendra Malik. Learning 3d human dynamics from video. In *CVPR*, pages 5614–5623, 2019.

[40] Rawal Khirodkar, Shashank Tripathi, and Kris Kitani. Occluded human mesh recovery. In *CVPR*, pages 1715–1725, June 2022.

[41] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *NeurIPS*, 2020.

[42] Muhammed Kocabas, Nikos Athanasiou, and Michael J Black. Vibe: Video inference for human body pose and shape estimation. In *CVPR*, pages 5253–5263, 2020.

[43] Muhammed Kocabas, Chun-Hao P. Huang, Otmar Hilliges, and Michael J. Black. Pare: Part attention regressor for 3d human body estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11127–11137, October 2021.

[44] Nikos Kolotouros, Georgios Pavlakos, Michael J Black, and Kostas Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *ICCV*, pages 2252–2261, 2019.

[45] Nikos Kolotouros, Georgios Pavlakos, and Kostas Daniilidis. Convolutional mesh regression for single-image human shape reconstruction. In *CVPR*, pages 4501–4510, 2019.

[46] Nikos Kolotouros, Georgios Pavlakos, Dinesh Jayaraman, and Kostas Daniilidis. Probabilistic modeling for human mesh recovery. In *ICCV*, pages 11605–11614, October 2021.

[47] Björn Krüger, Anna Vögele, Tobias Willig, Angela Yao, Reinhard Klein, and Andreas Weber. Efficient unsupervised temporal segmentation of motion data. *IEEE Transactions on Multimedia*, 2017.

[48] Bio Motion Lab. BMLhandball Motion Capture Database.

[49] Andreas Lehrmann, Peter V. Gehler, and Sebastian Nowozin. Efficient non-linear markov models for human motion. In *CVPR*, 2014.

[50] Jiefeng Li, Chao Xu, Zhicun Chen, Siyuan Bian, Lixin Yang, and Cewu Lu. Hybrik: A hybrid analytical-neural inverse kinematics solution for 3d human pose and shape estimation. In *CVPR*, pages 3383–3393, 2021.

[51] Maosen Li, Siheng Chen, Xu Chen, Ya Zhang, Yanfeng Wang, and Qi Tian. Actional-structural graph convolutional networks for skeleton-based action recognition. In *CVPR*, 2019.

[52] Wenhao Li, Hong Liu, Hao Tang, Pichao Wang, and Luc Van Gool. Mhformer: Multi-hypothesis transformer for 3d human pose estimation. In *Proceedings*

of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 13147–13156, 2022.

[53] Zhihao Li, Jianzhuang Liu, Zhensong Zhang, Songcen Xu, and Youliang Yan. Cliff: Carrying location information in full frames into human pose and shape estimation. In *ECCV*, 2022.

[54] Kevin Lin, Lijuan Wang, and Zicheng Liu. End-to-end human pose and mesh reconstruction with transformers. In *CVPR*, pages 1954–1963, 2021.

[55] Kevin Lin, Lijuan Wang, and Zicheng Liu. Mesh graphormer. In *ICCV*, pages 12939–12948, 2021.

[56] Lilang Lin, Sijie Song, Wenhan Yang, and Jiaying Liu. Ms2l: Multi-task self-supervised learning for skeleton based action recognition. In *ACM International Conference on Multimedia*, pages 2490–2498, 2020.

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

[58] Li Linguo, Wang Minsi, Ni Bingbing, Wang Hang, Yang Jiancheng, and Zhang Wenjun. 3d human action representation learning via cross-view consistency pursuit. In *CVPR*, 2021.

[59] Chunhui Liu, Yueyu Hu, Yanghao Li, Sijie Song, and Jiaying Liu. Pku-mmd: A large scale benchmark for continuous multi-modal human action understanding. *arXiv preprint arXiv:1703.07475*, 2017.

[60] Jun Liu, Amir Shahroudy, Mauricio Perez, Gang Wang, Ling-Yu Duan, and Alex C Kot. Ntu rgb+d 120: A large-scale benchmark for 3d human activity understanding. *IEEE TPAMI*, 42(10):2684–2701, 2019.

[61] Jun Liu, Amir Shahroudy, Dong Xu, Alex C Kot, and Gang Wang. Skeleton-based action recognition using spatio-temporal lstm network with trust gates. *IEEE TPAMI*, 40(12):3007–3021, 2017.

[62] Jun Liu, Gang Wang, Ping Hu, Ling-Yu Duan, and Alex C Kot. Global context-aware attention lstm networks for 3d action recognition. In *CVPR*, pages 1647–1656, 2017.

[63] 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*, 2020.

[64] Ziyu Liu, Hongwen Zhang, Zhenghao Chen, Zhiyong Wang, and Wanli Ouyang. Disentangling and unifying graph convolutions for skeleton-based action recognition. In *CVPR*, pages 143–152, 2020.

[65] Matthew Loper, Naureen Mahmood, and Michael J. Black. MoSh: Motion and Shape Capture from Sparse Markers. *ACM Transactions on Graphics*, 33(6), Nov. 2014.

[66] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM Transactions on Graphics*, 34(6):1–16, 2015.

[67] Eyes JAPAN Co. Ltd. Eyes Japan MoCap Dataset.

[68] Zhengyi Luo, S Alireza Golestaneh, and Kris M Kitani. 3d human motion estimation via motion compression and refinement. In *ACCV*, 2020.

[69] Diogo C. Luvizon, David Picard, and Hedi Tabia. 2d/3d pose estimation and action recognition using multitask deep learning. In *CVPR*, pages 5137–5146, 2018.

[70] Xiaoxuan Ma, Jiajun Su, Chunyu Wang, Wentao Zhu, and Yizhou Wang. 3d human mesh estimation from virtual markers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 534–543, June 2023.

[71] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. AMASS: Archive of motion capture as surface shapes. In *ICCV*, pages 5441–5450, Oct. 2019.

[72] Christian Mandery, Ömer Terlemez, Martin Do, Nikolaus Vahrenkamp, and Tamim Asfour. The kit whole-body human motion database. In *ICAR*, pages 329–336. IEEE, 2015.

[73] 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.

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

[75] Raphael Memmesheimer, Simon Häring, Nick Theisen, and Dietrich Paulus. Skeleton-dml: Deep metric learning for skeleton-based one-shot action recognition. In *WACV*, pages 3702–3710, 2022.

[76] Raphael Memmesheimer, Nick Theisen, and Dietrich Paulus. Sl-dml: Signal level deep metric learning for multimodal one-shot action recognition. In *ICPR*, 2021.

[77] Gyeongsik Moon, Juyong Chang, and Kyoung Mu Lee. Camera distance-aware top-down approach for 3d multi-person pose estimation from a single rgb image. In *ICCV*, 2019.

[78] Gyeongsik Moon and Kyoung Mu Lee. I2l-meshnet: Image-to-lixel prediction network for accurate 3d human pose and mesh estimation from a single rgb image. In *ECCV*, pages 752–768, 2020.

[79] Meinard Müller, Tido Röder, Michael Clausen, Bernhard Eberhardt, Björn Krüger, and Andreas Weber. Documentation mocap database HDM05. Technical Report CG-2007-2, Universität Bonn, June 2007.

[80] Mahdiar Nekoui and Li Cheng. Enhancing human motion assessment by self-supervised representation learning. In *BMVC*, 2021.

[81] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *ECCV*, 2016.

[82] Georgios Pavlakos, Xiaowei Zhou, and Kostas Daniilidis. Ordinal depth supervision for 3d human pose estimation. In *CVPR*, pages 7307–7316, 2018.

[83] 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.

[84] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 2020.

[85] Alberto Sabater, Laura Santos, Jose Santos-Victor, Alexandre Bernardino, Luis Montesano, and Ana C. Murillo. One-shot action recognition in challenging therapy scenarios. In *CVPR Workshop*, 2021.

[86] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In *CVPR*, pages 1010–1019, 2016.

[87] 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 *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part V*, pages 461–478. Springer, 2022.

[88] Dian Shao, Yue Zhao, Bo Dai, and Dahua Lin. Finegym: A hierarchical video dataset for fine-grained action understanding. In *CVPR*, pages 2616–2625, 2020.

[89] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In *CVPR*, 2019.

[90] Leonid Sigal, Alexandru O Balan, and Michael J Black. Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. *IJCV*, 87(1):4–27, 2010.

[91] Sijie Song, Cuiling Lan, Junliang Xing, Wenjun Zeng, and Jiaying Liu. An end-to-end spatio-temporal attention model for human action recognition from skeleton data. In *AAAI*, page 4263–4270, 2017.

[92] Yang Song, Luis Goncalves, and Pietro Perona. Unsupervised learning of human motion models. In T. Dietterich, S. Becker, and Z. Ghahramani, editors, *NeurIPS*, volume 14. MIT Press, 2001.

[93] Kun Su, Xiulong Liu, and Eli Shlizerman. Predict & cluster: Unsupervised skeleton based action recognition. In *CVPR*, pages 9631–9640, 2020.

[94] Yukun Su, Guosheng Lin, and Qingyao Wu. Self-supervised 3d skeleton action representation learning with motion consistency and continuity. In *ICCV*, 2021.

[95] Jiangxin Sun, Zihang Lin, Xintong Han, Jian-Fang Hu, Jia Xu, and Wei-Shi Zheng. Action-guided 3d human motion prediction. In *NeurIPS*, volume 34, pages 30169–30180. Curran Associates, Inc., 2021.

[96] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *CVPR*, 2019.

[97] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *ECCV*, 2018.

[98] Yu Sun, Qian Bao, Wu Liu, Yili Fu, Michael J Black, and Tao Mei. Monocular, one-stage, regression of multiple 3d people. In *ICCV*, pages 11179–11188, 2021.

[99] Yu Sun, Yun Ye, Wu Liu, Wenpeng Gao, Yili Fu, and Tao Mei. Human mesh recovery from monocular images via a skeleton-disentangled representation. In *ICCV*, pages 5349–5358, 2019.

[100] Fida Mohammad Thoker, Hazel Doughty, and Cees GM Snoek. Skeleton-contrastive 3d action representation learning. In *ACM International Conference on Multimedia*, pages 1655–1663, 2021.

[101] Dorra Trabelsi, Samer Mohammed, Faicel Chamroukhi, Latifa Oukhellou, and Yacine Amirat. An unsupervised approach for automatic activity recognition based on hidden markov model regression. *IEEE Transactions on Automation Science and Engineering*, 2013.

[102] Shashank Tripathi, Siddhant Ranade, Ambrish Tyagi, and Amit K. Agrawal. Posenet3d: Learning temporally consistent 3d human pose via knowledge distillation. In *8th International Conference on 3D Vision*, 2020.

[103] Nikolaus F. Troje. Decomposing biological motion: A framework for analysis and synthesis of human gait patterns. *Journal of Vision*, 2(5):2–2, Sept. 2002.

[104] Matthew Trumble, Andrew Gilbert, Charles Malleson, Adrian Hilton, and John P. Collomosse. Total capture: 3d human pose estimation fusing video and inertial sensors. In *BMVC*, 2017.

[105] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.

[106] Timo von Marcard, Roberto Henschel, Michael J Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *ECCV*, pages 601–617, 2018.

[107] Ziniu Wan, Zhengjia Li, Maoqing Tian, Jianbo Liu, Shuai Yi, and Hongsheng Li. Encoder-decoder with multi-level attention for 3d human shape and pose estimation. In *ICCV*, pages 13033–13042, 2021.

[108] Chunyu Wang, Yizhou Wang, and Alan L. Yuille. An approach to pose-based action recognition. In *CVPR*, pages 915–922, 2013.

[109] Jingbo Wang, Sijie Yan, Yuanjun Xiong, and Dahua Lin. Motion guided 3d pose estimation from videos. In *European Conference on Computer Vision*, pages 764–780. Springer, 2020.

[110] Peng Wang, Jun Wen, Chenyang Si, Yuntao Qian, and Liang Wang. Contrast-reconstruction representation learning for self-supervised skeleton-based action recognition. *arXiv preprint arXiv:2111.11051*, 2021.

[111] Xin Wang, Yudong Chen, and Wenwu Zhu. A survey on curriculum learning. *IEEE TPAMI*, 2022.

[112] Tom Wehrbein, Marco Rudolph, Bodo Rosenhahn, and Bastian Wandt. Probabilistic monocular 3d human pose estimation with normalizing flows. In *ICCV*, 2021.

[113] Wen-Li Wei, Jen-Chun Lin, Tyng-Luh Liu, and Hong-Yuan Mark Liao. Capturing humans in motion: Temporal-attentive 3d human pose and shape estimation from monocular video. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022.

[114] Tianhan Xu and Wataru Takano. Graph stacked hourglass networks for 3d human pose estimation. In *CVPR*, 2021.

[115] Yuanlu Xu, Song-Chun Zhu, and Tony Tung. Denserac: Joint 3d pose and shape estimation by dense render-and-compare. In *ICCV*, pages 7760–7770, 2019.

[116] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *AAAI*, 2018.

[117] Di Yang, Yaohui Wang, Antitza Dantcheva, Lorenzo Garattoni, Gianpiero Francesca, and Francois Bremond. Unik: A unified framework for real-world skeleton-based action recognition. In *BMVC*, 2021.

[118] Siyuan Yang, Jun Liu, Shijian Lu, Meng Hwa Er, and Alex C. Kot. Skeleton cloud colorization for unsupervised 3d action representation learning. In *ICCV*, 2021.

[119] Zhuoqian Yang, Wentao Zhu, Wayne Wu, Chen Qian, Qiang Zhou, Bolei Zhou, and Chen Change Loy. Transmomo: Invariance-driven unsupervised video motion retargeting. In *CVPR*, 2020.

[120] Angela Yao, Juergen Gall, and Luc Gool. Coupled action recognition and pose estimation from multiple views. *Int. J. Comput. Vision*, page 16–37, 2012.

[121] Chun-Han Yao, Jimei Yang, Duygu Ceylan, Yi Zhou, Yang Zhou, and Ming-Hsuan Yang. Learning visibility for robust dense human body estimation. In *European conference on computer vision (ECCV)*, 2022.

[122] Raymond Yeh, Yuan-Ting Hu, and Alexander Schwing. Chirality nets for human pose regression. In *NeurIPS*, 2019.

[123] Ailing Zeng, Lei Yang, Xuan Ju, Jiefeng Li, Jianyi Wang, and Qiang Xu. Smoothnet: A plug-and-play network for refining human poses in videos. In *European Conference on Computer Vision*. Springer, 2022.

[124] Feng Zhang, Xiatian Zhu, and Mao Ye. Fast human pose estimation. In *CVPR*, 2019.

[125] Hongwen Zhang, Yating Tian, Xinchi Zhou, Wanli Ouyang, Yebin Liu, Limin Wang, and Zhenan Sun. Pymaf: 3d human pose and shape regression with pyramidal mesh alignment feedback loop. In *ICCV*, pages 11446–11456, 2021.

[126] 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*, 2022.

[127] Ce Zheng, Sijie Zhu, Matias Mendieta, Taojiannan Yang, Chen Chen, and Zhengming Ding. 3d human pose estimation with spatial and temporal transformers. *ICCV*, 2021.

[128] Kun Zhou, Xiaoguang Han, Nianjuan Jiang, Kui Jia, and Jiangbo Lu. Hemlets pose: Learning part-centric heatmap triplets for accurate 3d human pose estimation. In *ICCV*, 2019.

[129] Wentao Zhu, Cuiling Lan, Junliang Xing, Wenjun Zeng, Yanghao Li, Li Shen, and Xiaohui Xie. Co-occurrence feature learning for skeleton based action recognition using regularized deep lstm networks. In *AAAI*, 2016.

[130] Wentao Zhu, Zhuoqian Yang, Ziang Di, Wayne Wu, Yizhou Wang, and Chen Change Loy. Mocanet: Motion retargeting in-the-wild via canonicalization networks. In *AAAI*, 2022.